

N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED
IN THE INTEREST OF MAKING AVAILABLE AS MUCH
INFORMATION AS POSSIBLE

NASA TECHNICAL MEMORANDUM

(NASA-TM-78292) LONGITUDINAL AND TRANSVERSE
MAGNETIC FIELD PROGRAM PROCEDURE AND
DETAILED SPECIFICATION FOR SIGMA 5 (NASA)
91 P HC A05/MF A01 CSCL 03B

N80-32314

G3/92 Unclass
28545

LONGITUDINAL AND TRANSVERSE MAGNETIC FIELD PROGRAM PROCEDURE AND DETAILED SPECIFICATION (FOR SIGMA 5)

By Caroline K. Wang
Space Sciences Laboratory

August 1980

NASA

*George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama*

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO. NASA TM-78292	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.	
4. TITLE AND SUBTITLE Longitudinal and Transverse Magnetic Field Program Procedure and Detailed Specification (For Sigma 5)		5. REPORT DATE August 1980	6. PERFORMING ORGANIZATION CODE
		8. PERFORMING ORGANIZATION REPORT #	
7. AUTHOR(S) Caroline K. Wang		10. WORK UNIT NO.	11. CONTRACT OR GRANT NO.
9. PERFORMING ORGANIZATION NAME AND ADDRESS George C. Marshall Space Flight Center Marshall Space Flight Center, Alabama 35812		13. TYPE OF REPORT & PERIOD COVERED Technical Memorandum	
		14. SPONSORING AGENCY CODE	
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration Washington, D.C. 20546			
15. SUPPLEMENTARY NOTES Prepared by Space Sciences Laboratory, Science and Engineering Directorate			
16. ABSTRACT This document presents a computer program and procedure for plotting the contour of the data transferred from the Marshall Space Flight Center solar magnetograph. The plotted data then can be easily compared with solar data from other sources, such as the Solar Maximum Mission (SMM).			
17. KEY WORDS		18. DISTRIBUTION STATEMENT <i>Caroline K. Wang</i> Unclassified—Unlimited	
19. SECURITY CLASSIF. (of this report) Unclassified	20. SECURITY CLASSIF. (of this page) Unclassified	21. NO. OF PAGES 95	22. PRICE NTIS

TABLE OF CONTENTS

	Page
1.0 PURPOSE	1
2.0 COMPUTER AND OPERATION SYSTEM	1
3.0 PROCEDURE	1
4.0 LONGITUDINAL PROGRAM FLOW DIAGRAM	3
5.0 TRANSVERSE PROGRAM FLOW DIAGRAM	5
APPENDIX A: SAMPLE CASES	7
APPENDIX B: PROGRAM LISTING	21
APPENDIX C: COMMAND FILES FOR KZTIME, KZLONGT, AND KZTRANSV	83
APPENDIX D: DESCRIPTIONS FOR PROGRAM SUBROUTINES	89

NASA TECHNICAL MEMORANDUM

1.0 PURPOSE

The purpose of the Longitudinal and Transverse programs is to plot the contour of the data which have been transferred from the Marshall Space Flight Center (MSFC) solar magnetograph. The data contain three different files:

1A1B For circular polarization

1A represents left circular; 1B represents right circular.

2A2B }
3A3B } For linear polarization

1.1 The Longitudinal program is to read the data 1A1B and plot the contours (1A - 1B/1A + 1B) for filtered longitudinal plot and intensity plot by choosing the positive and negative contour levels, intensity levels, and also X,Y plotting ranges which need to be used.

1.2 The Transverse program is to read the data files 2A2B and 3A3B and generate data

$$\sqrt{(2A - 2B/2A + 2B)^2 + (3A - 3B/3A + 3B)^2}$$

to plot the transverse contour plot, azimuth plot

$$\text{azimuth angle } \phi = \frac{1}{2} * \tan^{-1} \left[\frac{-(3A - 3B)/(3A + 3B)}{(2A - 2B)/(2A + 2B)} \right],$$

and intensity plot by choosing the positive and negative contour levels, intensity levels, and also X,Y plotting ranges which need to be used.

2.0 COMPUTER AND OPERATION SYSTEM

The program is operational under Sigma 5, using the Tektronics terminal.

3.0 PROCEDURE

3.1 Press Break key and press Return key. System will ask for ID number.

3.2 Key in WANG, ES52000 (ID will not show on the screen).

- 3.3 Key in PLATEN 132. (This will set the value of the terminal platen width.)
- 3.4 Key in XEQ KZTIME to execute KZTIME program which will list the housekeeping information from the header record of each file in the data tape. (An example is included in Appendix A.)

Key in PCL.

Key in REW FT#TEST (to rewind the tape).

Press Control key and Z key simultaneously.

- 3.5 Key in XEQ KZLONGT (to execute Longitudinal program).
An example of how to answer all the questions is included in Appendix A. The choice of record/data file equals the beginning record number of 1A1B file from KZTIME program output.
- 3.6 Key in XEQ KZTRANSV (to execute Transverse program).
An example is given in Appendix A. Record/data file equals the beginning record number of 2A2B file from KZTIME program output.
- 3.7 After running each program, key in PCL and then key in REW FT#TEST to rewind the tape.

Note: The following inputs should be used when a typing error occurs:

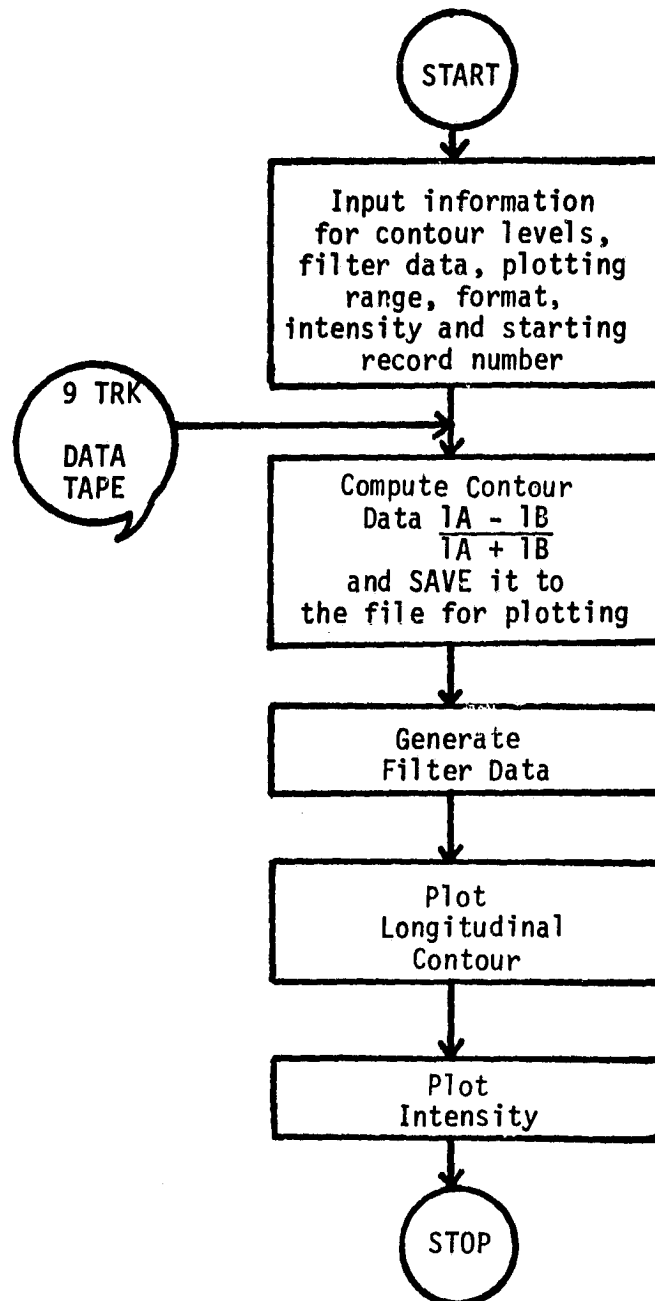
Hit "ROB" - Cancel one character typed on the line.

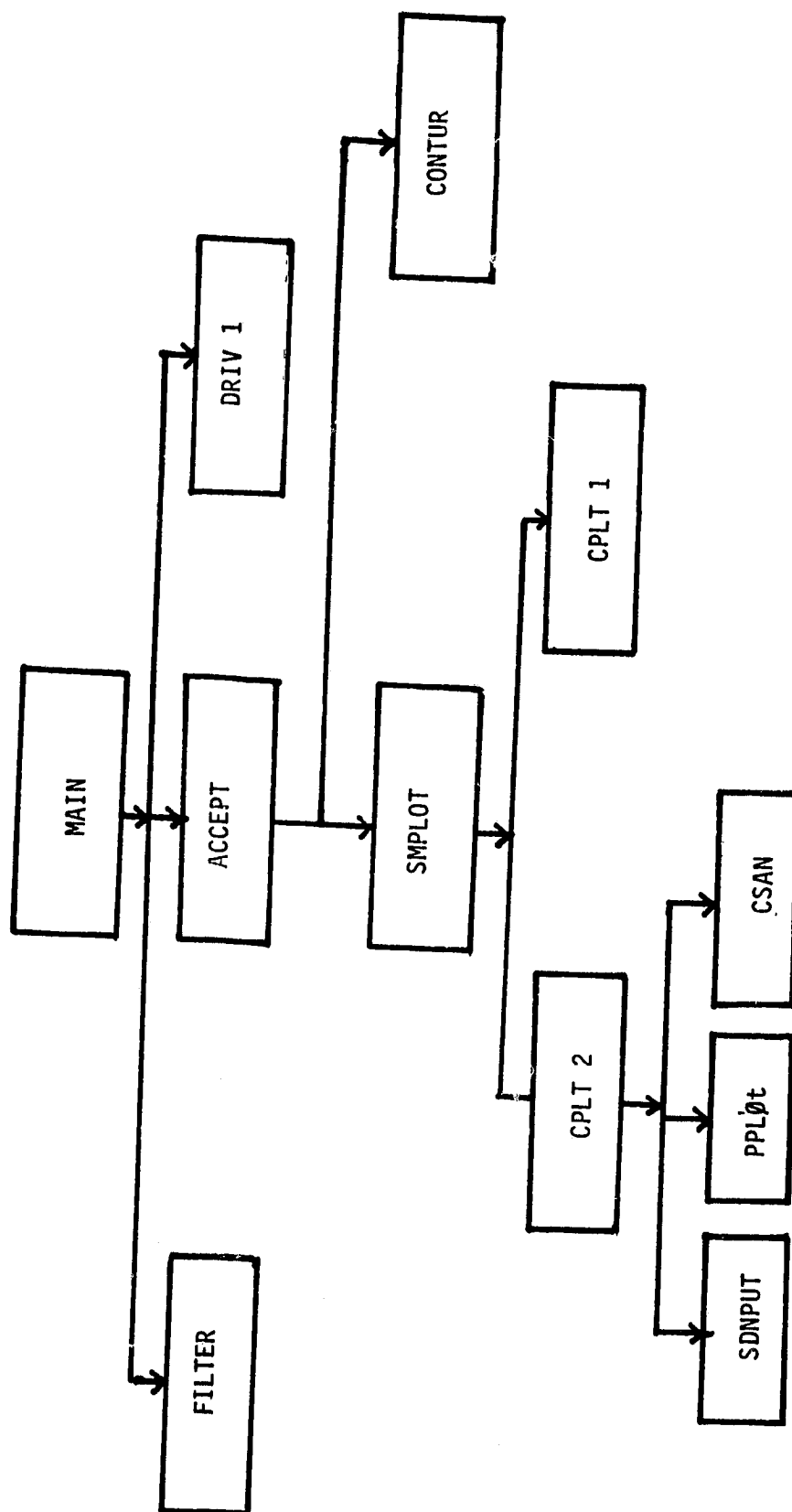
Hit "ESCAPE" and then hit "X" - Cancel all characters typed on the line.

Hit "ESCAPE" and another "ESCAPE" - Terminate the job.

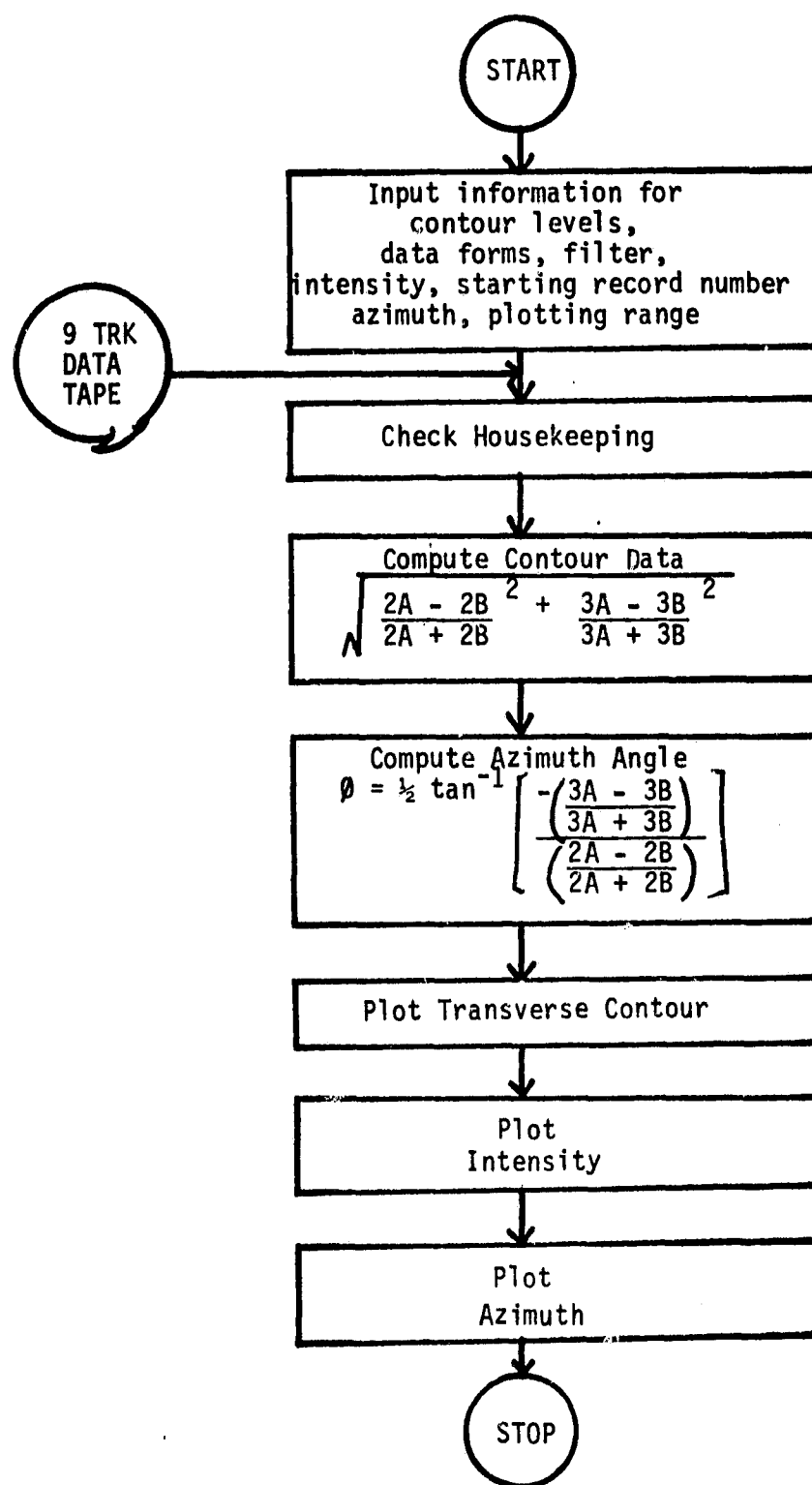
Every time a job is terminated, the tape must be rewound before running another program.

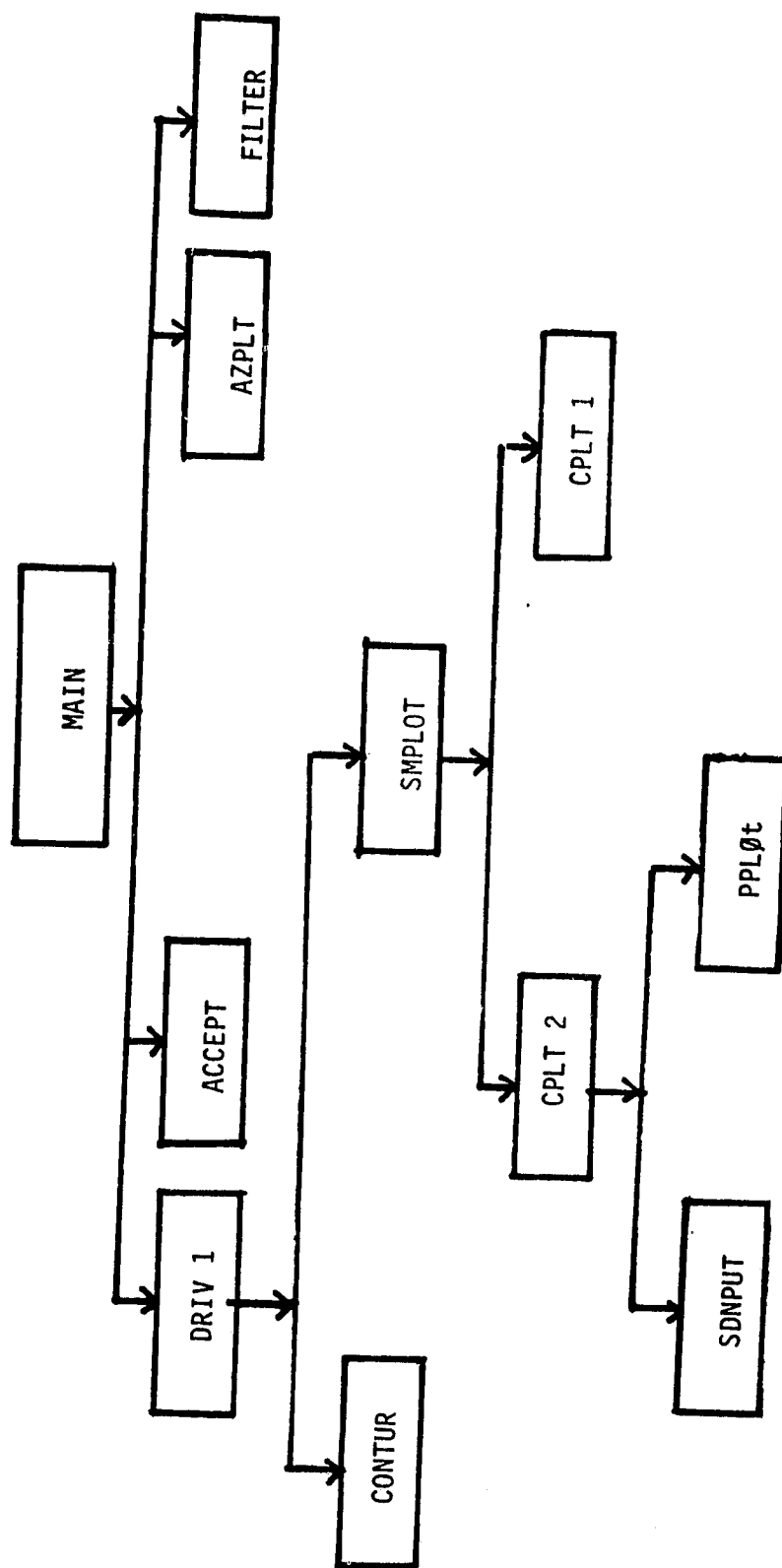
4.0 LONGITUDINAL PROGRAM FLOW DIAGRAM





5.0 TRANSVERSE PROGRAM FLOW DIAGRAM





APPENDIX A
SAMPLE CASES

Sample Case for KZTIME

```

XEO KXKZTIME
0001 - IECHO
0002 - IERROR
0003 - ISET F113 FTSTEST
0004 - ISET F110 ME
0005 - I LIKE TIME.
DO YOU WANT TO SKIP TO AN END OF FILE MARK (1=YE,0=NO)
?1
INPUT THE NUMBER OF EOF TO SKIP?
?1

EOF 1 RECORD COUNT 1
STARTING RECORD NUMBER?
?1

1 REC: 1 TIME:127:13:28:48 ZF:0870
ENH.:255 POL.:11 EXP.: 1/15
XPOS: .19 YPOS: 2.30

2 REC: 33 TIME:127:13:31:56 ZF:0870
ENH.:255 POL.:11 EXP.: 1/15
XPOS: .21 YPOS: 2.30

3 REC: 15 TIME:127:13:31:56 ZF:0870
ENH.:255 POL.:11 EXP.: 1/15
XPOS: .13 YPOS: 2.28

4 REC: 97 TIME:127:13:31:56 ZF:0870
ENH.:255 POL.:11 EXP.: 1/15
XPOS: .13 YPOS: 2.30

EOF REC.: 120

EOF REC.: 130

ESTOPS 0
XXXXX TERMINATEDXXX

```

ORIGINAL PAGE IS
OF POOR QUALITY

PAGE 8

10

[illegible]

ORIGINAL PAGE IS
OF POOR QUALITY

INTENSITY PLOT
(1A1B)

VALUE	LABEL
4000	A
3000	B
2000	C
1000	D
0	E
-1000	F
-2000	G
-3000	H
-4000	I

FILTERED LONGITUDINAL PLOT
TIME 187.1328148
ZEISS FILTER POSITION: 0.75 ENHANCEMENTS: 255
EXPOSURE: 1/15 POLARIZATION: 1
APERTURE: 10.120 VFO: 2.30
APERTURE: 10.120 V(10.120)

VALUE	LABEL	VALUE	VALUE
10	N	-10	
20	I	-20	
30	J	-30	
40	K	-40	
50	L	-50	
60	M	-60	
70	N	-70	
80		-80	
90		-90	
100		-100	
110		-110	
120		-120	
130		-130	
140		-140	
150		-150	
160		-160	
170		-170	
180		-180	
190		-190	
200		-200	
210		-210	
220		-220	
230		-230	
240		-240	
250		-250	



A- 74000
 B- 72500
 C- 73000
 D- 72500
 E- 72000
 F- 71500
 G- 71000
 WAS PROCESSING INTERRUPTED? (1=YES,0=NO)
 70 ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B
 (1=YES,0=NO)?
 70 NUMBER OF MAGNETOGRAMS TO PROCESS
 71 INPUT RECORD/DATA FILE NUMBERS OF MAGNETOGRAMS
 RECORD/DATA FILE
 72

DATA TO BE PLOTTED: 127:13:28:48
 NZFT-0870 NEIN- 255 NEXP-4 NPOL-1
 MIA(1)-D4481274 IDAT-1274D448

XE0 HUEZLONGT
 0001 - ECHO
 0002 - ERROR
 0003 - SET F113 FTEST
 0004 - SET F12 /10;OUTIN;SAVE;KEYED;DIRECT
 0005 - SET F13 /10;OUTIN;SAVE;KEYED;DIRECT
 0006 - SET F14 /10;OUTIN;SAVE;KEYED;DIRECT
 0007 - SET F15 /10;OUTIN;SAVE;KEYED;DIRECT
 0008 - SET F16 /10;OUTIN;SAVE;KEYED;DIRECT
 0009 - SET F17 /10;OUTIN;SAVE;KEYED;DIRECT
 0010 - SET F18 /10;OUTIN;SAVE;KEYED;DIRECT
 0011 - SET F19 /10;OUTIN;SAVE;KEYED;DIRECT
 0012 - SET F111 /10;OUTIN;SAVE;KEYED;DIRECT
 0013 - SET F114 LP
 0014 - SET F120 /INDS;OUT;SAVE
 0015 - SET F122 /INDS
 0016 - SET F110 ME
 0017 - SET F1101 ME
 0018 - HUEZLONGT.

NUMBER OF POSITIVE CONTOUR LEVELS
 77

A- 7400
 B- 7250
 C- 7300
 D- 7250
 E- 7200
 F- 7150
 G- 7100
 7250

DO YOU WANT THE POSITIVE AND NEGATIVE CONTOUR VALUES
 TO BE THE SAME (1=YES,0=NO)?
 71

FILTER DATA (1=YES,0=NO)?
 71

THERE ARE TWO DATA FORMATS.

1. 4.25X4.25 INCHES WHERE BOTH THE MAGNETOGRAM
 AND INTENSITY CAN BE PLACED ON SAME PAGE

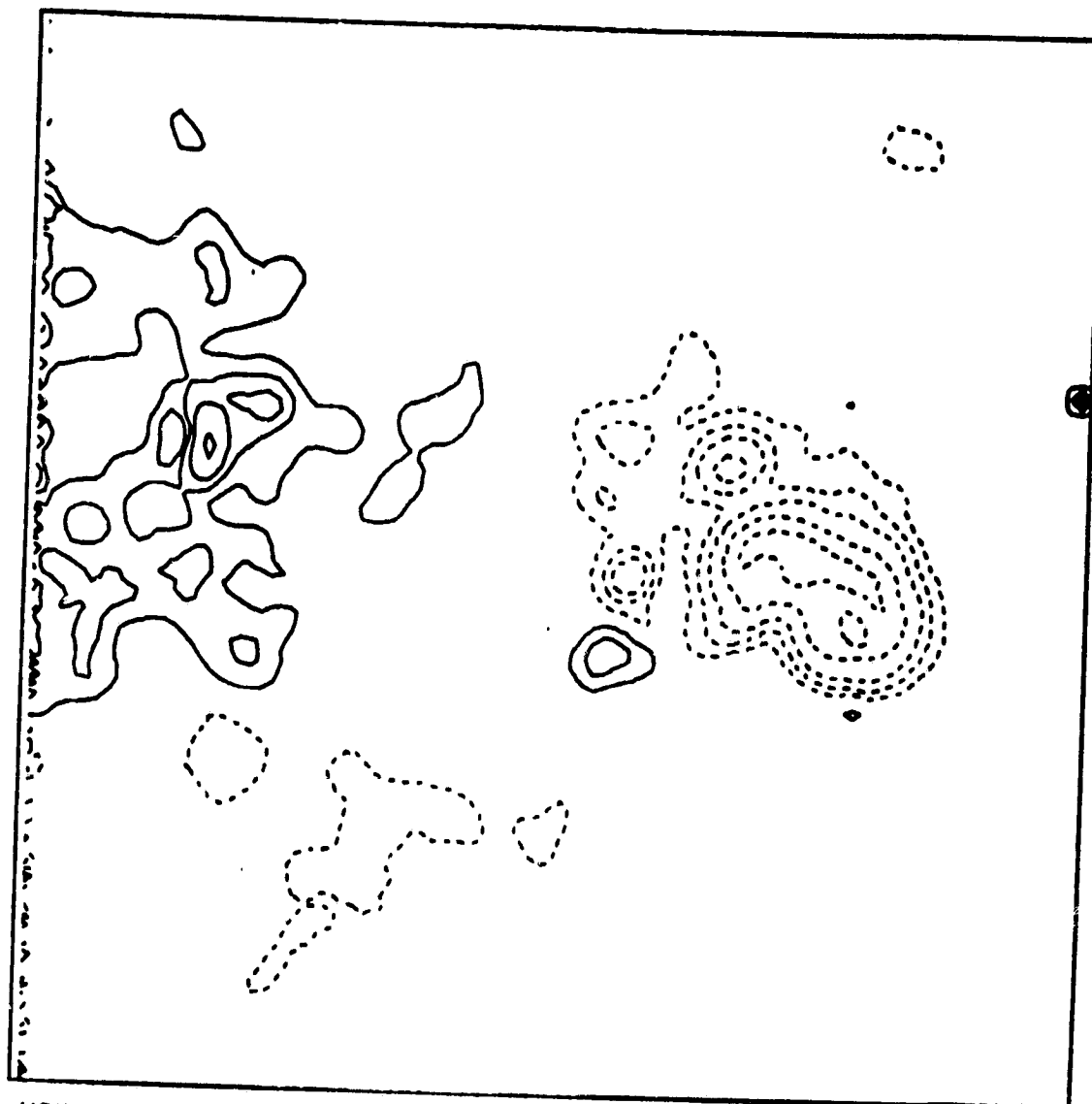
2. 6.5X8.5 INCHES ONLY ONE PLOT PER PAGE
 FORMAT (1 OR 2)?
 72

DELETE THE CHARACTER LABELS? (1=YES,0=NO)
 70

DELETE THE FIRST NEGATIVE AND POSITIVE CONTOUR LABELS(1=YES,0=NO) 77

XUTIN-
 710
 XMAX-
 7120
 YUTIN-
 710
 YMAX-
 7120
 DO YOU WANT INTENSITY PLOT? (1=YES,0=NO)
 71
 NUMBER OF INTENSITY VALUES (MAX. OF 7) 77

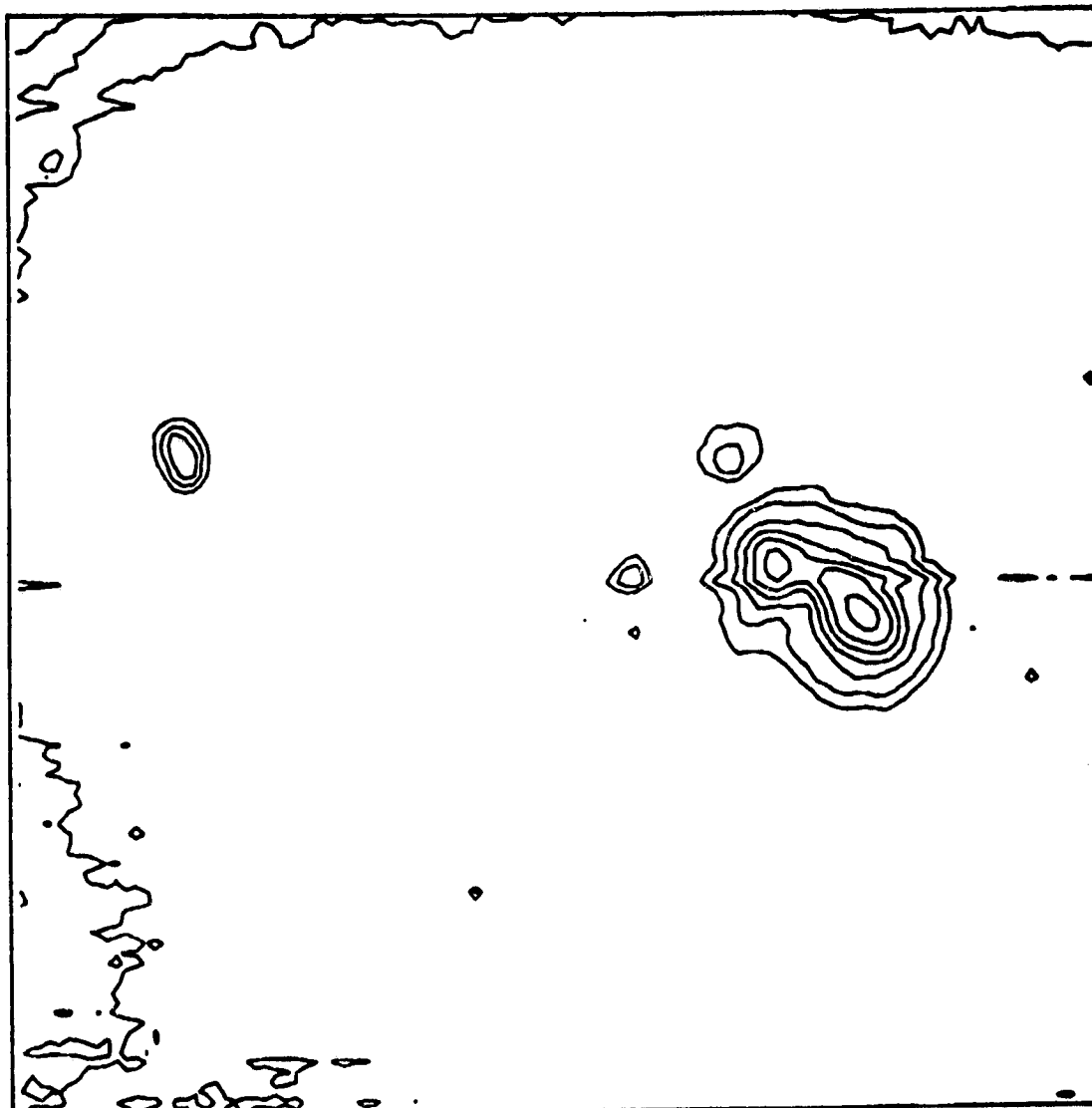
FILTERED LONGITUDINAL PLOT
 TIME 187:13:28:48
 Z100 FILTER POSITION: 0870
 EXPOSURE: 1/15 POLARIZATION: 1
 SPOT: 10 VPOS: 2.20
 APERTURE: (10,120) V(10,120)
 LABEL VALUE
 A B C D E F G
 10 25 50 100 150 200 250
 LABEL VALUE
 H I J K L M N
 -10 -25 -50 -100 -150 -200 -250



ORIGINAL PAGE IS
 OF POOR QUALITY

INTENSITY PLOT (2000)
 TIME 18713128148
 ZEISS FILTER POSITION: 0879 ENHANCEMENTS: 255
 EXPOSURE: 1/15 POLARIZATION: 1
 XPOS: .19 YPOS: 8.30
 APERTURE: X(10.120) Y(10.120)

VALUE	SCALE
4000	0
3500	1
3000	2
2500	3
2000	4
1500	5
1000	6



Sample Case for Transverse Program

```

XEO KICZTRANSDV
0001 - !ECHO
0002 - !ERROR
0003 - !SET F:13 FTEST
0004 - !SET F:12 /B:OUTIN;SAVE;KEYED;DIRECT
0005 - !SET F:13 /B:OUTIN;SAVE;KEYED;DIRECT
0006 - !SET F:14 /B:OUTIN;SAVE;KEYED;DIRECT
0007 - !SET F:15 /B:OUTIN;SAVE;KEYED;DIRECT
0008 - !SET F:16 /B:OUTIN;SAVE;KEYED;DIRECT
0009 - !SET F:17 /B:OUTIN;SAVE;KEYED;DIRECT
0010 - !SET F:18 /B:OUTIN;SAVE;KEYED;DIRECT
0011 - !SET F:19 /B:OUTIN;SAVE;KEYED;DIRECT
0012 - !SET F:11 /C:OUTIN;SAVE;KEYED;DIRECT
0013 - !SET F:12 /A:OUTIN;SAVE;KEYED;DIRECT
0014 - !SET F:20 /B:OUT;SAVE
0015 - !SET F:22 /B:OUT
0016 - !SET F:10 ME
0017 - !SET F:101 ME
0018 - !KICZTRANSDV.

TRANSVERSE PROGRAM

WAS DATA TO BE PLOTTED CREATED BEFORE APRIL 18,1980
(1=YES, 0=NO) ?
70

IF AZIMUTH PLOTS ONLY,SET NO. OF CONTOUR LEVELS
EQUAL TO ZERO
NUMBER OF CONTOUR LEVELS (MAX. NO. 7 ) ?
77
A=
75000
B=
74000
C=
73000
D=
72000
E=
71500
F=
71000
G=
7500

WAS PROCESSING INTERRUPTED (1=YES,0=NO)?
70

NUMBER OF MAGNETOGRAMS TO PROCESS?
71
ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B
(1=YES, 0=NO) ?
70

INPUT RECORD/DATA NUMBERS OF MAGNETOGRAMS
RECORD/DATA FILE .
734

DATA TO BE PLOTTED: 127:13:31:85
KZT-0270 NDMH- 225 NEDP-4 NPOL-1
RIA(1)-00001274 12MT-12740005

DO YOU WANT TO INPUT THE MIN. AND MAX. VALUES
FOR THE AZIMUTH PLOTS (1=YES,0=NO)?
70

DO YOU WANT INTENSITY PLOTS (1=YES,0=NO)?
71
2A,2B (1=YES,0=NO)?
71
3A,3B (1=YES,0=NO)?
71

NUMBER OF INTENSITY VALUES (MAX. NO. 7)?
77
INPUT INTENSITY VALUES DIVIDED BY 10
A=
75000
B=
74000
C=
73000
D=
72000
E=
71500
F=
71000
G=
7500

WAS PROCESSING INTERRUPTED (1=YES,0=NO)?
70

NUMBER OF MAGNETOGRAMS TO PROCESS?
71
ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B
(1=YES, 0=NO) ?
70

INPUT RECORD/DATA NUMBERS OF MAGNETOGRAMS
RECORD/DATA FILE .
734

DATA TO BE PLOTTED: 127:13:31:85
KZT-0270 NDMH- 225 NEDP-4 NPOL-1
RIA(1)-00001274 12MT-12740005

```

ORIGINAL PAGE IS
OF POOR QUALITY

MARSHALL SPACE FLIGHT CENTER PHOTOGRAPHY
TELEPHONE: 886-463-5667 FT 81578-5667

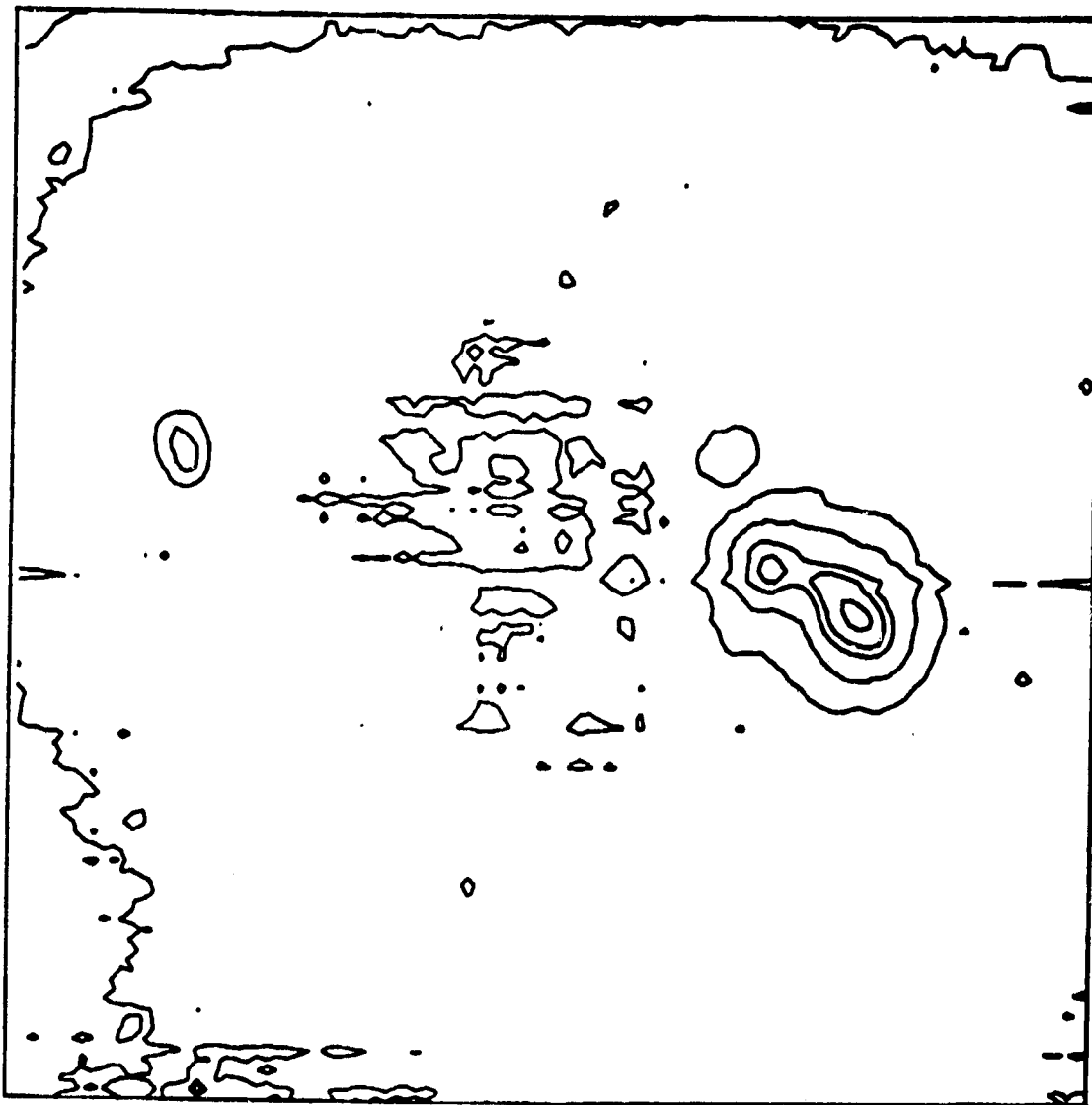
AZIMUTH PLOT

TIME: 1271131156
ZEISS FILTER: 0670 EXPOSURES: 255
EXPOSE: 1/15 POLARIZATIONS: 1
XPOS: 21 YPOS: 2.39
APERTURE: X(10, 120)
MINIMUM LENGTH SET AT 60.
MAXIMUM LENGTH SET AT 200.



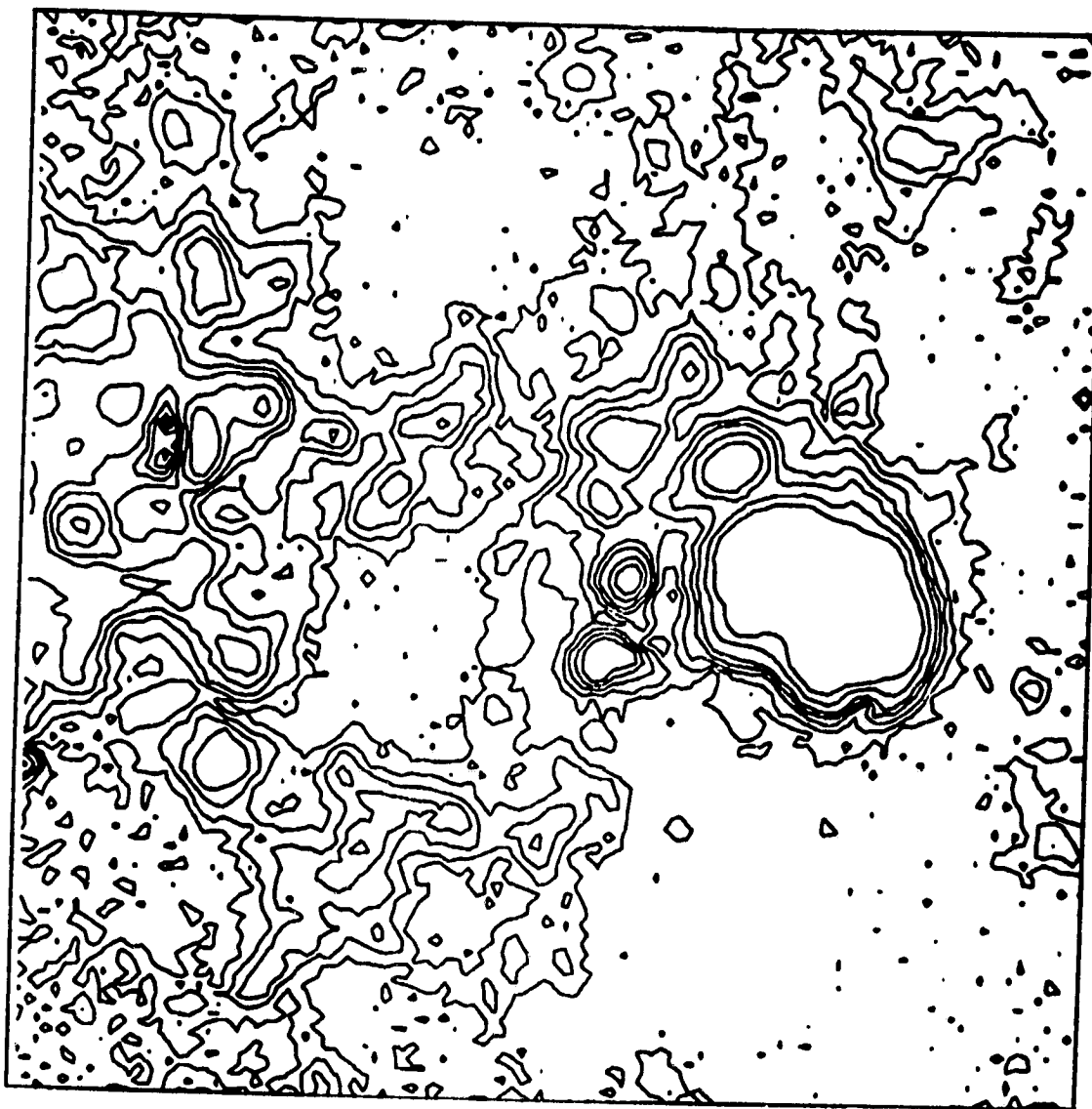
INTENSITY PLOT (2025)
 TIME: 1271313156
 ZEISS FILTER POSITION: 0870
 EXPOSURE: 1/15
 POLARIZATION: 1
 XPOS: -21
 YPOS: 2.30
 APERTURE: X(10.120)
 Y(10.120)

LABEL	VALUE
A	5000
B	4000
C	3000
D	2000
E	1500
F	1000
G	500



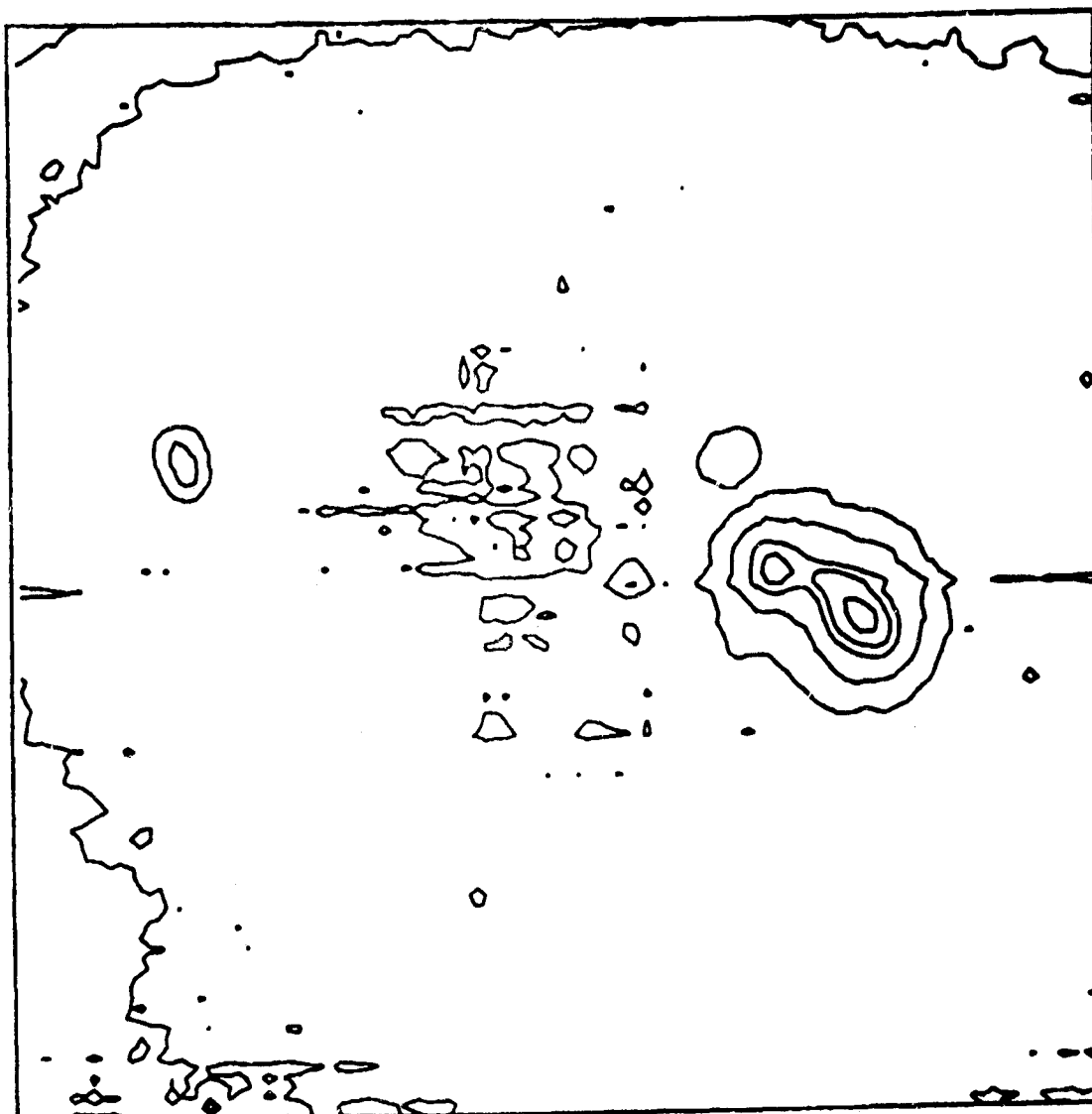
ORIGINAL PAGE IS
 OF POOR QUALITY

TRANSVERSE PLOT
TIME 127113:31:56
Z5155 FILTER POSITION: 0570 ENHANCEMENTS: 255
EXPOSURE: 1/16 POLARIZATIONS: 1
XPOS: .21 YPOS: 2.30
APERTURE: (10,120) V(10,120)



INTENSITY PLOT (3A3B)
 TIME 127:13:31.56
 ZEPHYR FILTER POSITION: 0879 ENHANCEMENTS: 255
 EXPOSURE: 1/15 POLARIZATION: 1
 XPOS: .21 YPOS: 2.30
 APERTURE: X(10,128) Y(10,128)

LABEL	VALUE
A	5000
B	4000
C	3000
D	2000
E	1500
F	1000
G	500



ORIGINAL PAGE IS
 OF POOR QUALITY

APPENDIX B
PROGRAM LISTING

PAGE 20 INTENTIONALLY BLANK

PROGRAM LISTING INDEX

NAME	PAGE
KZTIME	24
LONGT	27
DRV11	35
TRANSV	40
ACCEPT	49
FILTER	52
SMPL0T	55
CPLT1	58
CPLT2	60
SDNPUT	63
CSCAN	65
CONTUR	68
PTIME	80

EXT. FORTRAN IV, VERSION F02

ORIGINAL PAGE IS
OF POOR QUALITY

```

1. DIMENSION M00(15),M01(10),M02(8)
2. INTEGER DS
3. COMMON /HEAD/ INPBUF(512)
4. COMMON /IDAT/ DT(4),MFT,NEHH,NEXP,NPOL,XPOS,YPOS
5. DATA M00 /60H(1X,'DO YOU WANT TO SKIP TO AN END OF FILE MARK (1=YES
6. ,0=NO)'/
7. DATA M01 /40H(1X,'INPUT THE NUMBER OF EOF TO SKIP?') /
8. DATA M02 /32H(1X,'STARTING RECORD NUMBER?') /
9. C
10. C
11. C
12. 1000 FORMAT(1X,'EOF',I2,SX,'RECORD COUNT',I4)
13. 2000 FORMAT(1X,I4,' REC:',I5,2X,'TIME:',I3,3(' ',I2),6X,'ZF:',I2)
14. 3000 FORMAT(1X,'ENH:',I3,3X,'POL:',I1,3X,'EXP:',I1,I2)
15. 4000 FORMAT(1X,'XPOS:',F6.2,6X,'YPOS:',F6.2)
16. 5000 FORMAT(1X,'EOF',I2X,'REC',I1,I5)
17. 9991 FORMAT(1X,'INPUT TAPE ERROR , STATUS CODE =',I2)
18. C
19. C INPUT INFORMATION ABOUT END OF FILE MARK, STARTING RECORD NO.
20. C FROM TERMINAL
21. C
22. IREC=0
23. CALL ACCEPT (M00,IEOF)
24. IF(IEOF.EQ.0) GO TO 40
25. CALL ACCEPT (M01,NEOF)
26. DO 30 I=1,NEOF
27. IREC = I
28. 10 CALL BUFFERIN (13,1,INPBUF,512,ISTAT)
29. IF (ISTAT.EQ.3) GO TO 20
30. IREC = IREC + 1
31. GO TO 10
32. 20 WRITE(10,1000) I,IREC
33. 30 CONTINUE
34. 40 CONTINUE
35. CALL ACCEPT (M02,MSKIP)
36. MSKIP = MSKIP - 1
37. MP = 0
38. J = MSKIP
39. IF (MSKIP.EQ.0) GO TO 60
40. DO 50 I=1,MSKIP
41. CALL BUFFERIN (13,1,INPBUF,512,ISTAT)
42. 50 CONTINUE
43. 60 CONTINUE
44. IREC = IREC + 1
45. CALL BUFFERIN (13,1,INPBUF,512,ISTAT)
46. J = J + 1
47. IF( ISTAT.EQ.3) GO TO 90
48. IF(ISTAT.EQ.2) GO TO 70
49. WRITE(10,9991) ISTAT
50. 70 IF(IREC.EQ.1) GO TO 80
51. IF(IREC.EQ.32) IREC = 0
52. GO TO 60

```

```

53.      80  CONTINUE
54.      CALL PTIME
55.      NP = NP + 1
56.      OUTPUT = .
57.      WRITE(10,2000) NP,J,DT(4),DT(3),DT(2),DT(1),NZFT
58.      WRITE(10,3000) MEMH,PPQL,NBIP
59.      WRITE(10,4000) XPOS,YPOS
60.      GO TO 60
61.      90  CONTINUE
62.      OUTPUT = .
63.      WRITE(10,5000) J
64.      OUTPUT = .
65.      EOF1 = J
66.      IF(EOF1.EQ.EOF2) STOP
67.      EOF2 = EOF1 + 1
68.      IREC = 0
69.      GO TO 60
70.      END

```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC	HEX
WORDS	276	000E2
CONSTANTS	4	00004
LOCAL VARIABLES	44	0002C
TEMP	0	00000

TOTAL PROGRAM: 274 (PLUS LAYERS COMMON)

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

LONGT

```
1. C SUBROUTINE LONGT
2. C THIS ROUTINE SUBTRACTS THE 1A AND 1B MATRICES
3. C AND STORES THE RESULTS IN FILE "HL"
4. C WHICH IS USED BY SUBROUTINE DRVI TO PROCEDURE CONTOURS
5. C
6. C
7. C COMMON /BLNK/ NBT(4),MF(10),ID(4)
8. C COMMON /IVAL/ NCIR, KIRVAL(3),NMI,NCIRVAL(3),INTVAL(7)
9. C COMMON /BND/ XMIN,XMAX,YMIN,YMAX,IBIT
10. C COMMON /MCTAL/MCTAL(7),INUM
11. C COMMON /HL/MHL,HIL
12. C COMMON /ISA/ISA,INHOUS(8)
13. C COMMON /IDAT/IDT(4),NZFT,NEWH,NEXP,NPOL,XPOS,YPOS
14. C COMMON /ISB/ISB,NS,IDLCN,LCCH,LEXTRA,MFORM
15. C COMMON /IBIT/IBIT1,IBIT2,IBIT3,IBIT4,IBIT5,IBIT6,IBIT7
16. C COMMON /ABCA(500),G(500),X(130)
17. C INTEGER DT,INPBUF(512),MIA(64),MIB(64)
18. C REAL NML(128),MII(128)
19. C DATA NBT/7,7,6,10/
20. C DATA MF/1,2,4,8,10,20,40,80,100,200/
21. C DATA ID/227F,227F,223F,323FF/
22. C DATA KK/423FF/
23. C DATA MNM/1/
24. C DATA KZF/421FF/
25. C DATA KEH/22FF/
26. C DATA KMI1/82FFFFF0000/
27. C DATA KEXP/127/
28. C DIMENSION IRM(50)
29. C DIMENSION MN35(10),MN36(5),MN37(12),MN38(9)
30. C DIMENSION MN00(10),MN01(3),MN02(3),MN03(3),MN04(3),
31. C MN05(3),MN06(3),MN07(3),MN08(10),MN09(3),MN10(3),
32. C MN11(3),MN12(3),MN13(3),MN14(3),MN15(3),MN16(3),
33. C MN17(3),MN18(3),MN19(3),MN20(3),MN21(3),MN22(8),
34. C MN23(12),MN24(3),MN25(3),MN26(3),MN27(3),MN28(12),
35. C MN29(12),MN30(12),MN31(10),MN32(6),MN33(19),MN34(11)
36. C DATA MN00/40H(1),NUMBER OF POSITIVE CONTOUR LEVELS//
37. C DATA MN01/12H(1),A= //
38. C DATA MN02/12H(1),B= //
39. C DATA MN03/12H(1),C= //
40. C DATA MN04/12H(1),D= //
41. C DATA MN05/12H(1),E= //
42. C DATA MN06/12H(1),F= //
43. C DATA MN07/12H(1),G= //
44. C DATA MN08/40H(1),NUMBER OF NEGATIVE CONTOUR LEVELS//
45. C DATA MN09/12H(1),B= //
46. C DATA MN10/12H(1),C= //
47. C DATA MN11/12H(1),D= //
48. C DATA MN12/12H(1),E= //
49. C DATA MN13/12H(1),F= //
50. C DATA MN14/12H(1),G= //
51. C DATA MN15/12H(1),H= //
52. C DATA MN16/12H(1),I= //
```

```

28 53. DATA MM17/12H(1X,'J=- ')/
54. DATA MM18/12H(1X,'K=- ')/
55. DATA MM19/12H(1X,'L=- ')/
56. DATA MM20/12H(1X,'M=- ')/
57. DATA MM21/12H(1X,'N=- ')/
58. DATA MM22/32H(1X,'FILTER DATA (1=YES,0=NO)?')/
59. DATA MM23/48H(1X,'DELETE THE CHARACTER LABELS? (1=YES,0=NO)?')/
60. DATA MM24/12H(1X,'XMIN=- ')/
61. DATA MM25/12H(1X,'XMAX=- ')/
62. DATA MM26/12H(1X,'YMIN=- ')/
63. DATA MM27/12H(1X,'YMAX=- ')/
64. DATA MM28/48H(1X,'DO YOU WANT INTENSITY PLOTS? (1=YES,0=NO)?')/
65. DATA MM29/48H(1X,'NUMBER OF INTENSITY VALUES (MAX. OF 7)?')/
66. DATA MM30/48H(1X,'WAS PROCESSING INTERRUPTED? (1=YES,0=NO)?')/
67. DATA MM31/40H(1X,'NUMBER OF MAGNETOGRAMS TO PROCESS')/
68. DATA MM32/24H(1X,'FORMAT (1 OR 2)?')/
69. DATA MM33/76H(1X,'DELETE THE FIRST NEGATIVE AND POSITIVE CONTOURS L
70. LABELS(1=YES,0=NO)?')/
71. DATA MM34/44H(1X,'REPEAT LAST CONTOURS? (1=YES, 0=NO)?')/
72. DATA MM35/40H(1X,'INPUT FROM TAPE(1) OR DISK (0)?')/
73. DATA MM36/20H(1X,'(1=YES,0=NO)?')/
74. DATA MM37/28H(1X,'RECORD/DATE FILE = ')/
75. DATA MM38/36H(1X,'TO BE THE SAME (1=YES,0=NO)?')/
76. DO 10 I=1,7
77. MCTVAL(I)=0
78. MCTVAL(I)=0
79. INTVAL(I)=0
80. 10 CONTINUE
81. DO 20 I=1,50
82. 20 ICM(I)=0
83. DO 33 I=1,500
84. A(I)=0
85. B(I)=0
86. C(I)=0
87. 33 CONTINUE
88. DO 44 I=1,130
89. 44 Z(I)=0
90. DO 66 I=1,128
91. MHL(I)=0
92. HIL(I)=0
93. 66 CONTINUE
94. NF1=0
95. MKEOP=0
96. NEG=0
97. IBIT3=0
98. IBIT4=0
99. IBIT5=0
100. IBIT6=0
101. IBIT7=0
102. 1 CALL ACCEPT(MN00,NF1)
103. NCTR = NF1
104. NF = NF1

```

105. DO 4 I=1,NF
106. IF(I .EQ. 1) CALL ACCEPT(MM01,NCTVAL(I))
107. IF(I .EQ. 2) CALL ACCEPT(MM02,NCTVAL(I))
108. IF(I .EQ. 3) CALL ACCEPT(MM03,NCTVAL(I))
109. IF(I .EQ. 4) CALL ACCEPT(MM04,NCTVAL(I))
110. IF(I .EQ. 5) CALL ACCEPT(MM05,NCTVAL(I))
111. IF(I .EQ. 6) CALL ACCEPT(MM06,NCTVAL(I))
112. IF(I .EQ. 7) CALL ACCEPT(MM07,NCTVAL(I))
113. IF(I .GT. 7) OUTPUT 'ERROR = MAXIMUM OF 7 POSITIVE CONTOURS'
114. IF(I .GT. 7) GO TO 1
115. 4 CONTINUE
116. OUTPUT ' '
117. OUTPUT 'DO YOU WANT THE POSITIVE AND NEGATIVE CONTOUR VALUES'
118. CALL ACCEPT(MM30,ISM)
119. IF(ISM .EQ. 1) GO TO 6
120. OUTPUT ' '
121. 2 CALL ACCEPT(MM08,MM1)
122. DO 3 I=1,MM1
123. IF(NC .EQ. 1) OUTPUT 'NO POSITIVE VALUES'
124. IF(NC .EQ. 2) CALL ACCEPT(MM09,NCTVAL(I))
125. IF(NC .EQ. 3) CALL ACCEPT(MM10,NCTVAL(I))
126. IF(NC .EQ. 4) CALL ACCEPT(MM11,NCTVAL(I))
127. IF(NC .EQ. 5) CALL ACCEPT(MM12,NCTVAL(I))
128. IF(NC .EQ. 6) CALL ACCEPT(MM13,NCTVAL(I))
129. IF(NC .EQ. 7) CALL ACCEPT(MM14,NCTVAL(I))
130. IF(NC .EQ. 8) CALL ACCEPT(MM15,NCTVAL(I))
131. IF(NC .EQ. 9) CALL ACCEPT(MM16,NCTVAL(I))
132. IF(NC .EQ. 10) CALL ACCEPT(MM17,NCTVAL(I))
133. IF(NC .EQ. 11) CALL ACCEPT(MM18,NCTVAL(I))
134. IF(NC .EQ. 12) CALL ACCEPT(MM19,NCTVAL(I))
135. IF(NC .EQ. 13) CALL ACCEPT(MM20,NCTVAL(I))
136. IF(NC .EQ. 14) CALL ACCEPT(MM21,NCTVAL(I))
137. IF(NC .GT. 14) STOP
138. NCTVAL(I) = -NCTVAL(I)
139. IF(I .GT. 7) GO TO 2
140. NC = NC + 1
141. 3 CONTINUE
142. GO TO 9
143. 6 CONTINUE
144. MM1 = NF1
145. NEG = NF1
146. DO 7 I=1,NEG
147. 7 NCTVAL(I) = -NCTVAL(I)
148. 9 CONTINUE
149. OUTPUT ' '
150. CALL ACCEPT(MM22,IBIT3)
151. OUTPUT 'THERE ARE TWO DATA FORMATS.'
152. OUTPUT '1. 4.25X4.25 INCHES WHERE BOTH THE MAGNETOGRAM'
153. OUTPUT ' AND INTENSITY CAN BE PLACED ON SAME PAGE.'
154. OUTPUT ' '
155. OUTPUT '2. 6.5X6.5 INCHES ONLY ONE PLOT PER PAGE'
156. CALL ACCEPT(MM32,NFORM)

ORIGINAL PHOTO IS
OF POOR QUALITY

```

30 157. CALL ACCEPT (MM23,IDLCH)
158. CALL ACCEPT (MM33,LLCH)
159. CALL ACCEPT (MM24,XPIN)
160. CALL ACCEPT (MM25,RMAX)
161. CALL ACCEPT (MM26,YMIN)
162. CALL ACCEPT (MM27,RMAX)
163. CALL ACCEPT (MM28,IBIT5)
164. IF (IBIT5.EQ. 0) GO TO 35
165. CALL ACCEPT (MM29,INUM)
166. DO 34 I=1,INUM
167. IF (I.EQ. 1) CALL ACCEPT (MM01,INTVAL(I))
168. IF (I.EQ. 2) CALL ACCEPT (MM02,INTVAL(I))
169. IF (I.EQ. 3) CALL ACCEPT (MM03,INTVAL(I))
170. IF (I.EQ. 4) CALL ACCEPT (MM04,INTVAL(I))
171. IF (I.EQ. 5) CALL ACCEPT (MM05,INTVAL(I))
172. IF (I.EQ. 6) CALL ACCEPT (MM06,INTVAL(I))
173. IF (I.EQ. 7) CALL ACCEPT (MM07,INTVAL(I))
174. 34 CONTINUE
175. 35 CONTINUE
176. CALL ACCEPT (MM30,IBIT4)
177. IBIT6 = 0
178. IF (IBIT4.EQ. 1) CALL ACCEPT (MM34,IBIT6)
179. IF (IBIT4.EQ. 1) GO TO 16
180. OUTPUT MADE HERE FOR MARKS BETWEEN 1A18/2A28/3A38"
181. CALL ACCEPT (MM36,MKEOF)
182. CALL ACCEPT (MM31,NP)
183. OUTPUT "INPUT RECORD/DATA FILE NUMBERS OF MAGNETOGRAMS"
184. OUTPUT " "
185. DO 1000 I=1,NP
186. 1000 CALL ACCEPT (MM37,IRM(I))
187. 16 CONTINUE
188. C
189. C CREATE ZERO FILES
190. C
191. JREC=0
192. DO 4000 I3=1,NP
193. C
194. C MAG. TAPE INPUT
195. C
196. 130 IF (JREC.EQ. (IRM(I3)-1)) GO TO 120
197. CALL BUFFERIN(13,1,INPBUF,512,ISTAT)
198. JREC = JREC + 1
199. GO TO 130
200. 120 CONTINUE
201. DO 5 K=2,3
202. DO 110 I=1,16
203. CALL BUFFERIN(13,1,INPBUF,512,ISTAT)
204. IF (ISTAT.EQ. 2) GO TO 15
205. STOP
206. 15 CONTINUE
207. IF (I.GT. 1) GO TO 220
208. IREC = 0

```



```

209.      220 CONTINUE
210.      IWD = 1
211.      DO 45 I1 = 1, 8
212.      DO 50 J1 = 1, 64
213.      MIA(J1) = IMPBUF(IWD)
214.      IWD = IWD + 1
215.      50 CONTINUE
216.      CALL DIRECT WRITE (K, MIA, 64, IREG)
217.      IREC = IREC + 1
218.      45 CONTINUE
219.      JREC = JREC + 1
220.      110 CONTINUE
221.      5 CONTINUE
222.      C
223.      C READ END OF FILE
224.      C
225.      IF (NKEOF .EQ. 0) GO TO 55
226.      CALL BUFFERIN(13, 1, IMPBUF, 512, ISTAT)
227.      JREC = JREC + 1
228.      55 CONTINUE
229.      56 CONTINUE
230.      IRN3 = IRN(13)
231.      C
232.      C CHECK HOUSEKEEPING
233.      C
234.      CALL DIRECT READ (2, MIA, 64, 0)
235.      IDAT = ISL(MIA(1), -16) + IAND(ISL(MIA(1), 16), RMIA)
236.      DO 252 I1 = 1, 4
237.      IF (I1 .EQ. 1) NBB = 0
238.      IF (I1 .GT. 1) NBB = NBT(I1-1) + NBB
239.      DT(I1) = IAND(ID(I1), ISL(IDAT, -NBB))
240.      252 CONTINUE
241.      WRITE (10, 254) DT(4), DT(3), DT(2), DT(1)
242.      254 FORMAT(/, 1X, DATA TO BE PLOTTED: , 23, 3(1, 23))
243.      NZFT = IAND(ISL(MIA(2), -16), KZF)
244.      NENH = IAND(ISL(MIA(2), -8), KEN)
245.      NEXP = IAND(ISL(MIA(3), -25), KEXP)
246.      NPOL = IAND(ISL(MIA(3), -19), 7)
247.      WRITE(10, 264) NZFT, NENH, NEXP, NPOL
248.      264 FORMAT(1X, NZFT = , 24, 1X, NENH = , 15, 1X, NEXP = , 11, 1X, NPOL = , 11)
249.      WRITE(10, 274) MIA(1), IDAT
250.      274 FORMAT(1X, MIA(1) = , 28, 1X, IDAT = , 28)
251.      DO 25 I1 = 1, 8
252.      INHQS (I1) = MIA(I1)
253.      25 CONTINUE
254.      IF (NEXP .EQ. 3) NEXP = 60
255.      IF (NEXP .EQ. 3) NEXP = 30
256.      IF (NEXP .EQ. 4) NEXP = 15
257.      IF (NEXP .EQ. 5) NEXP = 8
258.      IAX = ISL(MIA(3), -16) + IAND(ISL(MIA(3), 16), RMIA)
259.      ISNGX = IAND(ISL(IX, -13), 1)
260.      NPUS = 0

```

ORIGINAL PAGE IS
OF POOR QUALITY

261.	TXPOS=0
262.	MX=14
263.	C DO 333 II=1,3
264.	II=3
265.	333 CONTINUE
266.	NN=4
267.	353 CONTINUE
268.	TXPOS=TXPOS+IAND(ISL(IXX,-MX),1)*(2**((NN-1)))
269.	NN = NN -1
270.	MX=MX+1
271.	IF(NN .GE. 1) GO TO 353
272.	XPOS=XPOS+TXPOS*(10**(II-1))
273.	TXPOS=0
274.	IF(II .LE. 1) GO TO 363
275.	II=II-1
276.	GO TO 333
277.	363 CONTINUE
278.	IF(ISNCY .EQ. 0) XPOS=XPOS
279.	ISNGY=IAND(ISL(IXX,-MX),1)
280.	YPOS=0
281.	YPOS=0
282.	MX=MX+1
283.	NN=4
284.	454 CONTINUE
285.	YPOS=YPOS+(IAND(ISL(IXX,-MX),1))*2**((NN-1))
286.	NN=NN-1
287.	MX=MX+1
288.	IF(NN .GE. 1) GO TO 454
289.	YPOS=YPOS+YPOS
290.	YPOS=YPOS*(10**(II-1))
291.	YPOS=0
292.	MV=23
293.	C DO 444 II=1,2
294.	II=2
295.	444 CONTINUE
296.	NN=4
297.	464 CONTINUE
298.	IYV=ISL(MIA(4),-16)+IAND(ISL(MIA(4),16),KMI1)
299.	YPOS=YPOS + (IAND(ISL(IYV,-MV),1))*2**((NN-1))
300.	NN=NN-1
301.	MV = MV + 1
302.	IF (NN .GE. 1) GO TO 464
303.	YPOS=YPOS + YPOS*(10**(II-1))
304.	YPOS=0
305.	C 444 CONTINUE
306.	IF(II .LE. 1) GO TO 474
307.	II=II-1
308.	GO TO 444
309.	474 CONTINUE
310.	IF(ISNCY .EQ. 0) YPOS=YPOS
311.	XPOS=XPOS/100.
312.	YPOS=YPOS/100.

```

313. C SUBTRACT MATRICES
314. C
315. C
316. I6OR = 2*10
317. C1 = 1000./ICOR
318. IYMIN=XYMIN
319. IXMAX=XMAX
320. IYMIN=XYMIN
321. IYMAX=YMAX
322. C INVERT YMIN 4 YMAX TO GET E-W DIRECTION CORRECT
323. C
324. DO 30 I=YMIN, IYMAX
325. IREC = I-1
326. CALL DIRECT READ (2,MIA,64,IREC)
327. CALL DIRECT READ (3,MIB,64,IREC)
328. DO 31 J=1,64
329. INPBUF (2*J-1) = ISL(MIA(J),-16)
330. INPBUF (2*J) = IAND(MIA(J),56)
331. INPBUF (128+2*J-1) = ISL(MIB(J),-16)
332. INPBUF (128+2*J) = IAND(MIB(J),56)
333. 31 CONTINUE
334. DO 40 J=YMIN, IYMAX
335. R1A = INPBUF (J)
336. R1B = INPBUF (128+J)
337. IF( R1A .GT. 0 .AND. R1B .GT. 0) GO TO 100
338. MHL(J)=0
339. GO TO 40
340. 100 MHL(J)=(R1A + R1B)/ICOR
341. MHL(J)=C1*((R1A - R1B)/MHL(J))
342. 40 CONTINUE
343. CALL DIRECT WRITE (6,MHL,128,IREC)
344. CALL DIRECT WRITE (7,MHL,128,IREC)
345. 30 CONTINUE
346. IF(1BIT3 .EQ. 1) CALL FILTER
347. 26 CONTINUE
348. C PLOT LONGITUDINAL CONTOURS
349. ISB = 1
350. IF(1BIT3 .EQ. 1) ISB = -1
351. NCTR = NF1
352. DO 90 IC=1, NCTR
353. 90 KTRVAL(IC) = NCTVAL(IC)
354. CALL DRIVE
355. IF(NFORM .EQ. 2 .OR. 1BIT5 .EQ. 0) CALL HDCOPY
356. IF(1BIT5 .EQ. 0) GO TO 36
357. C PLOT INTENSITY
358. ISB = 3
359. NCTR = INUM
360. DO 37 II=1, NCTR
361. 37 KTRVAL(II) = INTVAL(II)
362. CALL DRIVE
363. CALL HDCOPY
364. 36 CONTINUE

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

365.      CALL CLOSE(2)
366.      CALL CLOSE(3)
367.      CALL CLOSE(6)
368.      CALL CLOSE(7)
369.      IF (IRIT3.EQ. 1) CALL CLOSE(5)
370.      IF (IRIT4.EQ. 1 .OR. NP.EQ. 1) STOP
371.      4000 CONTINUE
372.      CALL FINITE(0,450)
373.      STOP
374.      END

```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

DEC	HEX
WORDS	WORDS

GENERATED CODE:	1271	004F7
CONSTANTS:	16	00010
LOCAL VARIABLES:	965	003C5
TEMPS:	3	00003

TOTAL PROGRAM: 2255 008CF (PLUS LABELED COMMON)

ORIGINAL PAGE IS
OF POOR QUALITY

DRVI1

```

1. C
2. C
3. SUBROUTINE DRVI1
4. COMMON /TVAL/ NCTR, KTRVAL(7), NM1, MCTVAL(7), INTVAL(7)
5. COMMON /BND/ XMIN, XMAX, YMIN, YMAX, IBIT
6. COMMON /CONCOM/ KFLAC1, KFLAC2, FUNCTR, NF, NDIR, NR, NY, NZ, X, Y, EXCLUD
7. COMMON /ISB/ ISB, NS, IDLCH, LLCH, LEXTRA, NFORM
8. COMMON /ABCJA(500), B(500), C(500), Z(130)
9. COMMON /MCTVAL/ MCTVAL(7), INOM
10. COMMON /HL/MHL, H1
11. COMMON /IA/IA
12. COMMON /IDAT/DT(4), NZFT, MEHH, NEXP, NPOL, XPOL, XPOS
13. COMMON /TSA/TSA(60), INHOURS(8)
14. COMMON /IBIT/IBIT1, IBIT2, IBIT3, IBIT4, IBIT5, IBIT6, IBIT7
15. INTEGER ALPHA(14), DT
16. REAL HL(128), H1(128)
17. DIMENSION CVAL(14)
18. DATA ALPHA/1HA, 1HB, 1HC, 1HD, 1HE, 1HF, 1HG, 1HH, 1HI, 1HJ, 1HK, 1HL, 1HM,
19. 1HN/
20. DATA NMN /1/
21. IA=0
22. IF (ISB.EQ. -1) KU=5
23. IF (ISB.EQ. 1) KU=6
24. IF (ISB.EQ. 2) KU=6
25. IF (ISB.EQ. 3) KU=7
26. IF (ISB.EQ. 4) KU=7
27. IF (ISB.EQ. 5) KU=8
28. IF (ISB.GT. 5) GR, ISB=17, -1) STOP
29. IF (NCTR.EQ. 0 .AND. ISB.EQ. 2) GO TO 1100
30. C
31. C PAUSE AT DRVI1
32. IPASS=0
33. YMIN1 = YMIN
34. YMAX1 = YMAX
35. NX = XMAX - XMIN + 1
36. NY = YMAX - YMIN + 1
37. YMAX= 129. - YMIN1
38. YMIN= 129. - YMAX1
39. C
40. C CHECK TO SEE IF PLOT IS LONG.
41. C
42. NDIR=500
43. EXCLUD=-1
44. C
45. C POSITIVE CONTOUR VALUES
46. C
47. IF (ISB.GE. 3) GO TO 103
48. DO 5 I=1, NCTR
49. CVAL(I)=KTRVAL(I)
50. A(I)=MCTVAL(I)
51. 5 CONTINUE
52. NC=NCTR

```

```

53. C      NEGATIVE CONTOUR VALUES
54. C
55. C
56. MSG=MM1
57. IF(NEC.EQ.0) GO TO 104
58. DO 6 I=1,MSG
59. NC=NC+1
60. A(NG)=NCTVAL(I)
61. 6 CONTINUE
62. 104 CONTINUE
63. NF=NC
64. GO TO 102
65. 103 DO 7 I=1,INUM
66. A(I)=INTVAL(I)
67. 7 CONTINUE
68. NF=INUM
69. 102 CONTINUE
70. IRECP=128-YMIN+1
71. KFLAG1=1
72. IRII=1
73. 11 CONTINUE
74. NNN=MM+1
75. CALL CONTUR
76. 988 GO TO (21,22,22,22,24),KFLAG2
77. 21 IX=X
78. 22 IV=Y
79. IREC=128-IV
80. IF(IREC.EQ.IRECP) GO TO 10
81. CALL DIRECT READ (KU,HL,128,IREC)
82. IRECP=IRECP-1
83. 10 FUNCTN=HL(IX)
84. GO TO 11
85. 22 CONTINUE
86. IPASS=IPASS+1
87. CALL DIRECT WRITE (12,A,500,IA)
88. CALL DIRECT WRITE (9,B,500,IA)
89. CALL DIRECT WRITE (11,C,500,IA)
90. IA=IA+1
91. GO TO (99,11,99),KFLAG2
92. 24 WRITE(10,95)
93. 95 FORMAT(1X,31H CONTUR NEEDS MORE BUFFER SPACE)
94. 99 CONTINUE
95. 40 CONTINUE
96. CALL SMPLOT
97. CALL RESTAT(TSA)
98. C      CALL NCNVT(300,30,XMAX)
99. C      CALL NCNVT(950,30,XMIN)
100. C      CALL NCNVT(270,50,YMAX)
101. C      CALL NCNVT(270,750,YMIN)
102. CALL MOVABS(0,767)
103. CALL APYODE
104. C      PRINT CONTOUR LEVELS

```

```

105. IF(NFORM.EQ.1.AND.ISB.GT.2)GO TO 2000
106. IF(ISB.EQ.-1)OUTPUT-'FILTERED-LONGITUDINAL PLOT'-
107. IF (ISB.EQ. 1) OUTPUT-'TRANSVERSE PLOT'-
108. IF (ISB.EQ. 2) OUTPUT-'INTENSITY PLOT (I1I8) '-
109. IF (ISB.EQ. 3) OUTPUT-'INTENSITY PLOT (2A2B) '-
110. IF (ISB.EQ. 4) OUTPUT-'INTENSITY PLOT (3A3B) '-
111. WRITE(10,300) DT(4),DT(3),DT(2),DT(1)
112. FORMAT(1X,'TIME',Z3,'-',Z2,'-',Z2,'-',Z2)
113. WRITE(10,301) NZFT,NENH
114. FORMAT(1X,'ZEISS-FILTER-POSITION',Z4,Z3,'ENHANCEMENT',Z5,Z4)
115. WRITE(10,303) NEXP,NPOL
116. 303 FORMAT(1X,'EXPONENT',I7,'-',I2,4X,'POLARIZATION',I3)
117. WRITE(10,309) XPOS,YPOS
118. 309 FORMAT(1X,'XPOS',F6-2,I5X,'YPOS',F6-2)
119. WRITE(10,313) XMIN,XMAX,YMIN,YMAX
120. 313 FORMAT(1X,'APERTURE',I7,'-',I3,'-',I3,'-',I3,'-',I3,'-',I3)
121. IF (ISB.GT.2)GO TO 314
122. IF (NEG.EQ.0) WRITE(10,305)
123. IF (NEG.GT.0) WRITE(10,306)
124. DO 200 I=1,NCTR
125. NC=I+7
126. INC=I*NCTR
127. IF (NEG.EQ.0) WRITE(10,307) ALPHA(I),MCTVAL(I)
128. IF (NEG.GT.0) WRITE(10,308) ALPHA(I),MCTVAL(I),ALPHA(ING),MCTVAL(1)
129. 200 CONTINUE
130. GO TO 3000
131. 314 CONTINUE
132. WRITE(10,305)
133. DO 315 I=1,INUM
134. WRITE(10,307) ALPHA(I),INTVAL(I)
135. 315 CONTINUE
136. GO TO 3000
137. 2000 CONTINUE
138. 310 FORMAT(1X,66X,'INTENSITY PLOT')
139. WRITE(10,310)
140. IF (ISB.EQ. 3) WRITE(10,323)
141. IF (ISB.EQ. 4) WRITE(10,324)
142. IF (ISB.EQ. 5) WRITE(10,325)
143. 323 FORMAT(67X,'(I1I8)')
144. 324 FORMAT(67X,'(2A2B)')
145. 325 FORMAT(67X,'(3A3B)')
146. WRITE(10,311)
147. DO 201 I=1,NCTR
148. WRITE(10,312) ALPHA(I),INTVAL(I)
149. 311 FORMAT(10X,'ALPHA',4X,'VALUE')
150. 312 FORMAT(69X,A1,6X,I5)
151. 201 CONTINUE
152. 3000 CONTINUE
153. C
154. PAUSE BEFORE SWAP PLOT.SV IN DRV1.SV
155. 305 FORMAT(1X,'LABEL',4X,'VALUE')
156. 306 FORMAT(1X,'LABEL',4X,'VALUE',8X,'LABEL',4X,'VALUE')
157. 307 FORMAT(3X,A1,6X,I5)

```

ORIGINAL PAGE IS
OF POOR QUALITY

EXT. FORTRAN IV, VERSION F02

PRECEDING PAGE BLANK NOT FILMED

Program TRANSV

```

1. COMMON /TVAL/NCIR,KIRVAL(7),NNI,NCTVAL(7),INTVAL(7)
2. COMMON /ABNO/AMIN,AMAX
3. COMMON /ISB/ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
4. COMMON /IDAT/IDT(4),NREF,NSNH,NEXP,NPDB,XPBS,YPOS
5. COMMON /BND/XMIN,XMAX,YMIN,YMAX,IBIT
6. COMMON /NCTVAL/NCTVAL(7),INUM
7. COMMON /ABC/A(500),B(500),C(500),Z(130)
8. COMMON /BLNK/BLNK(4),NF(10),ID(4)
9. COMMON /TSA/TSA(60),INHOURS(8)
10. INTEGER M2A(128),M2B(128),M3A(128),M3B(128),IMP8UP(512)
11. INTEGER DT
12. DIMENSION IRM(50)
13. DIMENSION M00(12),M01(3),M02(3),M03(3),M04(3),M05(3)
14. DIMENSION M06(3),M07(3),M08(3),M09(3),M10(3),M11(3)
15. DIMENSION M12(3),M13(3),M14(3),M15(11),M16(8),M17(8)
16. DIMENSION M18(12),M19(7),M20(7),M21(12),M22(12)
17. DIMENSION M23(11),M24(10),M25(11),M26(6),M27(7)
18. DATA M01/2,7,6,10/
19. DATA M02/1,2,4,8,10,20,40,80,100,200/
20. DATA M03/2276,3276,3236,3236/
21. DATA M04/42FFFF/
22. DATA M05/82FFFF0000/
23. DATA M06/421FFF/
24. DATA M07/22FE/
25. DATA M08/127/
26. DATA M09/48H(1X,"NUMBER OF CONTOUR LEVELS (MAX. NO. 7) ? ?")/
27. DATA M10/12H(1X,"A=")/
28. DATA M11/12H(1X,"B=")/
29. DATA M12/12H(1X,"C=")/
30. DATA M13/12H(1X,"D=")/
31. DATA M14/12H(1X,"E=")/
32. DATA M15/12H(1X,"F=")/
33. DATA M16/12H(1X,"G=")/
34. DATA M17/56H(1X,"FORMAT (1 OR 2) ?")/
35. DATA M18/56H(1X,"DELETE ALL OF THE CHARACTER LABELS? (1=YES,0=NO) ?")/
36. DATA M19/52H(1X,"DELETE THE FIRST CONTOUR LEVEL? (1=YES,0=NO) ?")/
37. DATA M20/12H(1X,"XMIN=")/
38. DATA M21/12H(1X,"XMAX=")/
39. DATA M22/12H(1X,"YMIN=")/
40. DATA M23/12H(1X,"YMAX=")/
41. DATA M24/44H(1X,"FOR THE AZIMUTH PLOTS (1=YES,0=NO) ?")/
42. DATA M25/32H(1X,"MAXIMUM VALUE (EX. 200) ?")/
43. DATA M26/32H(1X,"MINIMUM VALUE (EX. 60) ?")/
44. DATA M27/48H(1X,"DO YOU WANT INTENSITY PLOTS (1=YES,0=NO) ?")/
45. DATA M28/28H(1X,"3A,3B (1=YES,0=NO) ?")/
46. DATA M29/48H(1X,"NUMBER OF INTENSITY VALUES (MAX. NO. 7) ?")/
47. DATA M30/48H(1X,"WAS PROCESSING INTERRUPTED (1=YES,0=NO) ?")/
48. DATA M31/44H(1X,"REPEAT LAST CONTOUR? (1=YES,0=NO) ?")/
49. DATA M32/40H(1X,"INPUT FROM TAPE (1) OR DISK (0) ?")/
50. DATA M33/44H(1X,"NUMBER OF MAGNETOGRAMS TO PROCESS ?")/
51. DATA M34/44H(1X,"NUMBER OF MAGNETOGRAMS TO PROCESS ?")/

```

ORIGINAL PAGE IS
OF POOR QUALITY

```
53. DATA M26/24H(1X,'(1=YES, 0=NO) ?') /
54. DATA M27/28H(1X,'RECORD/DATA FILE =') /
55. C IBIT1 : 2A28
56. C IBIT2 : 2A38
57. C IBIT3 : DATA CREATED BEFORE APRIL 18, 1980
58. C IBIT4 : HAS PROCESSING INTERRUPTED
59. C IBIT5 : INTENSITY PLOT
60. DO 10 I=1,7
61. MCTVAL(I)=0
62. MCTVAL(I)=0
63. 10 INTVAL(I)=0
64. IBIT1=0
65. IBIT2=0
66. IBIT3=0
67. IBIT4=0
68. IBIT5=0
69. IBIT6=0
70. IBIT7=0
71. NEG=0
72. MKEOF=0
73. ISB=0
74. DO 20 I=1,50
75. 20 IRN(I)=0
76. OUTPUT ..
77. OUTPUT ..
78. OUTPUT 'TRANSVERSE PROGRAM'
79. OUTPUT ..
80. OUTPUT ..
81. 17 CONTINUE
82. OUTPUT 'HAS DATA TO BE PLOTTED CREATED BEFORE APRIL 18, 1980'
83. C CORRECT SEQUENCE SWITCH(2A28 NOW 3A38)
84. CALL ACCEPT(M26,IBIT3)
85. IF(IBIT3.EQ.0.OR. IBIT3.EQ.1) GO TO 18
86. GO TO 17
87. 18 CONTINUE
88. OUTPUT ..
89. OUTPUT 'IF AZIMUTH PLOTS ONLY, SET NO. OF CONTOUR LEVELS'
90. OUTPUT 'EQUAL TO ZERO'
91. CALL ACCEPT(M00,NF1)
92. IF(NF1.EQ.0) GO TO 90
93. NF=NF1
94. MCTR=NF1
95. DO 4 I=1,NF
96. IF(I.EQ.1) CALL ACCEPT(M01,MCTVAL(I))
97. IF(I.EQ.2) CALL ACCEPT(M02,MCTVAL(I))
98. IF(I.EQ.3) CALL ACCEPT(M03,MCTVAL(I))
99. IF(I.EQ.4) CALL ACCEPT(M04,MCTVAL(I))
100. IF(I.EQ.5) CALL ACCEPT(M05,MCTVAL(I))
101. IF(I.EQ.6) CALL ACCEPT(M06,MCTVAL(I))
102. IF(I.EQ.7) CALL ACCEPT(M07,MCTVAL(I))
103. 4 CONTINUE
104. 90 CONTINUE
```

```

105. OUTPUT ..
106. OUTPUT THERE ARE TWO DATA FORMATS.
107. OUTPUT 1. 4.25=4.25 INCHES WHERE BOTH THE MAGNETOGRAM
108. OUTPUT AND INTENSITY CAN BE PLACED ON THE SAME PAGE.
109. OUTPUT ..
110. OUTPUT 2. 6.5=6.5 INCHES ONLY ONE PLOT PER PAGE.
111. CALL ACCEPT (M08,MFORM)
112. OUTPUT ..
113. CALL ACCEPT(M09,IDLCH)
114. OUTPUT ..
115. CALL ACCEPT(M10,LLCH)
116. OUTPUT ..
117. OUTPUT ENTER ARRAY SIZE (MAXIMUM 128*128)
118. CALL ACCEPT(M11,XMIN)
119. CALL ACCEPT(M12,XMAX)
120. CALL ACCEPT(M13,YMIN)
121. CALL ACCEPT(M14,YMAX)
122. OUTPUT ..
123. OUTPUT DO YOU WANT TO INPUT THE MIN. AND MAX. VALUES
124. CALL ACCEPT (M15,IAZ)
125. IF(IAZ.EQ. 1) GO TO 15
126. AMAX = 200.
127. AMIN = 60.
128. GO TO 16
129. 15 CONTINUE
130. CALL ACCEPT (M16,AMAX)
131. CALL ACCEPT (M17,AMIN)
132. 16 CONTINUE
133. OUTPUT ..
134. CALL ACCEPT(M18,IBIT5)
135. IF(IBIT5.EQ. 0) GO TO 80
136. CALL ACCEPT(M19,IBIT1)
137. CALL ACCEPT(M20,IBIT2)
138. OUTPUT ..
139. CALL ACCEPT(M21,INUM)
140. OUTPUT INPUT INTENSITY VALUES DIVIDED BY 10
141. DO 81 I=1,INUM
142. IF(I.EQ. 1) CALL ACCEPT (M01,INTVAL(I))
143. IF(I.EQ. 2) CALL ACCEPT (M02,INTVAL(I))
144. IF(I.EQ. 3) CALL ACCEPT(M03,INTVAL(I))
145. IF(I.EQ. 4) CALL ACCEPT(M04,INTVAL(I))
146. IF(I.EQ. 5) CALL ACCEPT(M05,INTVAL(I))
147. IF(I.EQ. 6) CALL ACCEPT(M06,INTVAL(I))
148. IF(I.EQ. 7) CALL ACCEPT(M07,INTVAL(I))
149. 81 CONTINUE
150. 80 CONTINUE
151. OUTPUT ..
152. CALL ACCEPT(M22,IBIT4)
153. IBIT6 = 0
154. OUTPUT ..
155. C IF(IBIT4.EQ. 1) CALL ACCEPT(M23,IBIT6)
156. C IF(IBIT4.EQ. 1) GO TO 26

```

ORIGINAL PAGE IS
OF POOR QUALITY

```
157.      OUTPUT .
158.      CALL ACCEPT(M25,NP)
159.      OUTPUT 'ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B'
160.      CALL ACCEPT(M26,MREQF)
161.      OUTPUT .
162.      OUTPUT 'INPUT RECORD/DATA NUMBERS OF MAGNETOGRAMS'
163.      DO 1000 I=1,NP
164.      1000 CALL ACCEPT(M27,IRN(I))
165.      C
166.      C SET UP INPUT FILES
167.      C
168.      26 CONTINUE
169.      C
170.      C MAG. TAP INPUT
171.      C
172.      JREC=0
173.      DO 4000 I3=1,NP
174.      4000 IF(JREC.EQ.(IRN(I3)-1)) GO TO 120
175.      CALL BUFFERIN(I3,1,INBUF,512,ISTAT)
176.      JREC=JREC+1
177.      GO TO 130
178.      120 CONTINUE
179.      DO 5 K=2,5
180.      DO 11 I=1,16
181.      60 CALL BUFFERIN(I3,1,INBUF,512,ISTAT)
182.      IF(I.GT.1) GO TO 201
183.      IREC=0
184.      201 CONTINUE
185.      IWD=1
186.      DO 45 J1=1,8
187.      DO 50 J1=1,64
188.      M2A(J1)=INBUF(IWD)
189.      IWD=IWD+1
190.      50 CONTINUE
191.      CALL DIRECT WRITE(K,M2A,64,IREC)
192.      IREC=IREC+1
193.      45 CONTINUE
194.      JREC=JREC+1
195.      11 CONTINUE
196.      5 CONTINUE
197.      C
198.      C CHECK HOUSEKEEPING
199.      C
200.      CALL DIRECT READ (2,M2A,64,0)
201.      IDAT=ISL(M2A(1),-16)*IAND(ISL(M2A(1),16),RM1A)
202.      DO 252 I1=1,4
203.      IF (I1.EQ.1) MRB = 0
204.      IF (I1.GT.1) MRB = MRB(I1-1) + MRB
205.      DT(I1) = IAND(ID(I1),ISL(IDAT,-MRB))
206.      252 CONTINUE
207.      WRITE (10,254) DT(4),DT(3),DT(2),DT(1)
208.      254 FORMAT(//1X,'DATA TO BE PLOTTED',33,3(' ',22))
```

```

209. NZFT=IAND(ISL(M2A(2),-16),KZF)
210. MENH=IAND(ISL(M2A(2),-8),KSH)
211. NEXP=IAND(ISL(M2A(3),-25),KEXP)
212. NPOL=IAND(ISL(M2A(3),-19),77)
213. WRITE(10,264) NZFT,MENH,NEXP,NPOL
214. 264 FORMAT(1X,'NZFT=',24,1X,'MENH=',15,1X,'NEXP=',11,1X,'NPOL=',11)
215. WRITE(10,274) M2A(1),IDAT
216. 274 FORMAT(1X,'M2A(1)=' ,2B,1X,'IDAT=' ,2B)
217. DO 25 I1=1,8
218. INHOU5(I1) = M2A(I1)
219. 25 CONTINUE
220. IF(NEXP.EQ.3) NEXP=60
221. IF(NEXP.EQ.3) NEXP=30
222. IF(NEXP.EQ.4) NEXP=15
223. IF(NEXP.EQ.5) NEXP=8
224. IF(NEXP.EQ.6) NEXP=4
225. IXX =ISL(M2A(3),-16)+IAND(ISL(M2A(3),16),KM1A)
226. ISNGX=IAND(ISL(IXX,-13),1)
227. XPOS=0
228. TXPOS=0
229. MX=14
230. C DO 333 I1=1,3
231. I1=3
232. 333 CONTINUE
233. NN=4
234. 353 CONTINUE
235. TXPOS=TXPOS+IAND(ISL(IXX,-MX),1)*(2**((NN-1)))
236. NN = NN -1
237. MX=MX+1
238. IF(NN.GE.1) GO TO 353
239. XPOS=XPOS+TXPOS*(10**((I1-1)))
240. TXPOS=0
241. IF(I1.LE.1) GO TO 363
242. I1=I1-1
243. GO TO 333
244. 363 CONTINUE
245. IF(ISNGX.EQ.0) XPOS=-XPOS
246. ISNGY=IAND(ISL(IXX,-MX),1)
247. YPOS=0
248. TYPOS=0
249. MY=MX+1
250. NN=4
251. 454 CONTINUE
252. TYPOS=TYPOS+(IAND(ISL(IXX,-MX),1))*2**((NN-1))
253. NN=NN-1
254. MY=MY+1
255. IF(NN.GE.1) GO TO 454
256. YPOS=YPOS+TYPOS
257. YPOS=YPOS*(10**2)
258. TYPOS=0
259. MY=23
260. C DO 444 I1=1,2

```

```

261.      II=2
262.      444 CONTINUE
263.      NN=4
264.      464 CONTINUE
265.      IYV=ISL(M2A(4),-16)+IAND(ISL(M2A(4),16),KM1A)
266.      YPOS=YPOS + (IAND(ISL(IYV,-MT),1)) * 2 ** (NN-1)
267.      NN=NN - 1
268.      NY = NY + 1
269.      IF (NN .GE. 1) GO TO 464
270.      YPOS=YPOS + YPOS*(10**(II-1))
271.      YPOS=0
272.      G 444 CONTINUE
273.      IF (II .LE. 1) GO TO 474
274.      II=II-1
275.      GO TO 444
276.      474 CONTINUE
277.      IF (ISNGY .EQ. 0) YPOS=-YPOS
278.      XPOS=XPOS/100.
279.      YPOS=YPOS/100.
280.      C *****
281.      C *****
282.      C *****
283.      C SUBTRACT MATRICES
284.      C
285.      ICUR=2*10
286.      UOBAR=0.
287.      ROBAR=0.
288.      K2=1
289.      C2=1000.
290.      IYMIN=YMIN
291.      IYMAX=YMAX
292.      IXMIN=XMIN
293.      IXMAX=XMAX
294.      DO 30 I=IYMIN,IYMAX
295.      IREC=I-1
296.      CALL DIRECT READ(2,M2A,64,IREC)
297.      CALL DIRECT READ(3,M2B,64,IREC)
298.      CALL DIRECT READ(4,M3A,64,IREC)
299.      CALL DIRECT READ(5,M3B,64,IREC)
300.      C CONVERT 16-BIT INTEGERS TO 32-BIT INTEGERS
301.      DO 31 J=1,64
302.      ITEMP=M2A(65-J)
303.      M2A(139-2*J)=IAND(ITEMP,KK)
304.      M2A(129-2*J)=ISL(ITEMP,-16)
305.      ITEMP=M2B(65-J)
306.      M2B(139-2*J)=IAND(ITEMP,KK)
307.      M2B(129-2*J)=ISL(ITEMP,-16)
308.      ITEMP=M3A(65-J)
309.      M3A(139-2*J)=IAND(ITEMP,KK)
310.      M3A(129-2*J)=ISL(ITEMP,-16)
311.      ITEMP=M3B(65-J)
312.      M3B(139-2*J)=IAND(ITEMP,KK)

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

46 313. M3B(129-2*J)=ISL(ITEMP,-16)
314. 31 CONTINUE
315. DU 40 J=IXMIN,IXMAX
316. R2A=M2A(J)
317. R2B=M2B(J)
318. IF(R2A-CT. 0.0 .AND. R2B-CT. 0.0) GO TO 100
319. U=0
320. GO TO 43
321. 100 U=((R2A-R2B)/(R2A+R2B))-U0BAR
322. 43 R3A=M3A(J)
323. R3B=M3B(J)
324. IF(R3A-CT. 0.0 .AND. R3B-CT. 0.0) GO TO 1001
325. R=0
326. GO TO 44
327. 1001 R=((R3A-R3B)/(R3A+R3B))-R0BAR
328. 44 A(J)=C2*((U**2+R**2)**.25)
329. B(J)=(R3A+R3B)/ICOR
330. C(J)=(R2A+R2B)/ICOR
331. C TEST WHEN DATA WAS TAKEN
332. C
333. IF(I8IT3.EQ. 1) GO TO 39
334. Z(J)=-45
335. IF(R.EQ. 0. .AND. U.GE. 0.) GO TO 40
336. Z(J)=45
337. IF(R.EQ. 0. .AND. U.LT. 0.) GO TO 40
338. W=U/R
339. Z(J)=5*ATAN(W)*(180./3.14159)
340. IF(U.GE. 0. .AND. R.LT. 0.) Z(J)=-90+Z(J)
341. IF(U.LT. 0. .AND. R.LT. 0.) Z(J)=Z(J)+90
342. GO TO 40
343. 39 CONTINUE
344. Z(J)=-45
345. IF(U.EQ. 0.0 .AND. R.GE. 0.0) GO TO 40
346. Z(J)=45
347. IF(U.EQ. 0.0 .AND. R.LT. 0.0) GOTO 40
348. W=R/U
349. Z(J)=5*ATAN(W)*(180./3.14159)
350. IF(R.GE. 0.0 .AND. U.LT. 0.0) Z(J)=-90+Z(J)
351. IF(R.LT. 0.0 .AND. U.LT. 0.0) Z(J)=Z(J)+90
352. 40 CONTINUE
353. KREC=IREC+128
354. C TRANSVERSE PLOT IS8=2
355. CALL DIRECT WRITE(6,A,128,IREC)
356. C AZPLOT IS8=1
357. CALL DIRECT WRITE(6,Z,128,KREC)
358. C 2A2B PLOT IS8=4
359. CALL DIRECT WRITE(7,C,128,IREC)
360. C 3A3B PLOT IS8=5
361. CALL DIRECT WRITE(8,B,128,IREC)
362. 30 CONTINUE
363. C
364. C PLOT AZIMUTH PLOTS

```

```

365. C CALL AZPLT
366. IF(NFORM.EQ. 2 .OR. IBIT1.EQ. 0 .OR. IBIT5.EQ. 0) CALL HOCOPY
368. C PLOT 2A2B INTENSITY
369. C
370. C IF(IBIT5.EQ. 0 .OR. IBIT1.EQ. 0) GO TO 98
371. ISB=4
372. CALL DRIV1
373. CALL HOCOPY
374. C
375. C PLOT TRANSVERSE CONTOURS
376. C
377. C 98 CONTINUE
378. ISB=2
379. IF(NCTR.EQ. 0) GO TO 92
380. CALL DRIV1
381. IF(NFORM.EQ. 2 .OR. IBIT5.EQ. 0 .OR. IBIT2.EQ. 0) CALL HOCOPY
382. C
383. C 92 CONTINUE
384. C
385. C PLOT 3A3B INTENSITY
386. C
387. ISB=5
388. IF(IBIT5.EQ. 0 .OR. IBIT2.EQ. 0) GO TO 99
389. CALL DRIV1
390. CALL HOCOPY
391. C
392. C 99 CONTINUE
393. IF(IBIT4.EQ. 1 .OR. MP.EQ. 1) STOP
394. 4000 CONTINUE
395. CALL FINITI(0,450)
396. STOP
END

```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC	HEX
GENERATED CODE:	1365	00555
CONSTANTS:	28	0001C
LOCAL VARIABLES:	1333	00535
TEMPS:	3	00003

TOTAL PROGRAM: 2729 00AA9 (PLUS LABELED COMMON)

ORIGINAL PAGE IS
OF POOR QUALITY

ACCEPT

```

1. SUBROUTINE ACCEPT(IAR,IDUM)
2. DIMENSION IAR(1)
3. WRITE(101,IAR)
4. READ(101,10)IDUM
5. 10 FORMAT(I)
6. RETURN
7. END
    
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	18	00012
CONSTANTS:	0	00000
LOCAL VARIABLES:	1	00001
TEMPS:	3	00003
TOTAL PROGRAM:	22	00016

ORIGINAL PAGE 11
OF POOR QUALITY

50

```
1. SUBROUTINE ACCEPTR(IAR,XDUM)
2. DIMENSION IAR(1)
3. WRITE(101,IAR)
4. READ(101,10)XDUM
5. 10 FORMAT(F)
6. RETURN
7. END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	18	00012
CONSTANTS:	0	00000
LOCAL VARIABLES:	1	00001
TEMPS:	3	00003
TOTAL PROGRAM:	22	00016

ORIGINAL IMAGE IS
OF POOR QUALITY

FILTER

```

52 1. SUBROUTINE FILTER
2. COMMON /BND/ SWIN,XAY,YMIN,YMAX,IBIT
3. INTEGER WF1,WF2,WF3
4. REAL FHL(128),MHL(384),MHL2,MHL3
5. EQUIVALENCE (MHL2,MHL(129)),(MHL3,MHL(257))
6. IXMIN=IWIN
7. IXMAX=XMAX
8. IYMIN=YMIN-1
9. IYMAX=YMAX
10. WF1=20
11. WF2=3
12. WF3=2
13. TW=WF1+4*WF2+4*WF3
14. * THE FOPEN FUNCTION WILL BE ACHIEVED VIA
15. * ISET F:5 /FHL,OUTIN,SAVE,KEYED,DIRECT
16. * CALL FOPEN(5,FHL,512)
17. DO 10 I=1,128
18. FHL(I)=0
19. C ZERO EDGES OF FILTERED MAGNETOGRAM
20. * CALL WRITR(5,0,FHL,1,IER)
21. * CALL WRITR(5,127,FHL,1,IER)
22. * CALL DIRECT WRITE (5,FHL,128,0)
23. * CALL DIRECT WRITE (5,FHL,128,127)
24. C READ 3 LINES OF DATA FOR AVERAGING
25. * PRIME THE INPUT BUFFERS WITH THE FIRST TWO RECORDS
26. * CALL DIRECT READ (6,MHL,128,0)
27. * CALL DIRECT READ (6,MHL2,128,1)
28. DO 20 IR=IYMIN,IYMAX
29. * IREC=IR-1
30. IREC=IR+1
31. * CALL READR(6,IREC,MHL,3,IER)
32. * CALL DIRECT READ (6,MHL3,128,IREC)
33. FHL(1)=0
34. FHL(128)=0
35. DO 30 J=2,127
36. IC1=J-1
37. IC2=J
38. IC3=J+1
39. IC4=J+127
40. IC5=J+128
41. IC6=J+129
42. IC7=J+255
43. IC8=J+256
44. IC9=J+257
45. A=WF1*MHL(IC5)
46. B=WF2*(MHL(IC2)+MHL(IC4)+MHL(IC6)+MHL(IC8))
47. C=WF3*(MHL(IC1)+MHL(IC3)+MHL(IC7)+MHL(IC9))
48. FHL(J)=(A+B+C)/TW
49. 30 CONTINUE
50. * CALL WRITR(5,IR,FHL,1,IER)
51. * CALL DIRECT WRITE (5,FHL,128,IR)
52. * IN-CORE SHUFFLE TO PREPARE FOR NEXT READ

```

53. DO 15 J=1,256
 54. 15 MHL(J) = MHL(J+128)
 55. 20 CONTINUE
 56. * THE CP-V IMPLEMENTATION CLOSURES P+5 IN LONGP
 57. * CALL FCLOS(5)
 58. RETURN
 59. END

ORIGINAL PAGE IS
 OF POOR QUALITY

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	163	00043
CONSTANTS:	7	00007
LOCAL VARIABLES:	539	00219
TEMPS:	2	00002
TOTAL PROGRAM:	711	002C7 (PLUS LABELED COMMON)

TEXT. FORTRAN IV, VERSION F02

ORIGINAL PAGE IS
OF POOR QUALITY

SMPLOT

```

1. C SMPLOT.SV
2. C THIS ROUTINE CALLS EITHER CPLO11 OR CPLO12 AFTER
3. C BEING SWAPPED
4. C SUBROUTINE SMPLOT
5. COMMON /TVAL,NCTR,KTRVAL(7),NN1,NCTVAL(7),INTVAL(7)
6. COMMON /ISB,ISB,N6,IDLCH,SLCH,LSHRA,NFORM
7. COMMON /BND,XMIN,XMAX,YMIN,YMAX,IBIT
8. COMMON /ARGA(500),B(500),C(500),Z(130)
9. COMMON /TSA/TSA(60),INHOUS(8)
10. COMMON /IA/IA
11. YMIN1=YMIN
12. YMAX1=YMAX
13. YMIN=129.-YMAX1
14. YMAX=129.-YMIN1
15. IF(NFORM.NE.1) GO TO 10
16. IF(NFORM.EQ.1.AND.ISB.LE.2) GO TO 10
17. GO TO 20
18. 10 CALL INITT(960)
19. CALL NEWPAG
20. GO TO 30
21. 20 CALL RESTAT(TSA)
22. 30 CONTINUE
23. CALL CPLT1
24. IF (ISB.EQ.2.AND.NCTR.EQ.0) GO TO 1001
25. DO 1000 IA2=1,IA
26. IP=IA2-1
27. CALL DIRECT READ (12,A,500,IP)
28. CALL DIRECT READ (9,B,500,IP)
29. CALL DIRECT READ (11,C,500,IP)
30. C
31. C SET FOR SOLID CONTOURS, CSCAN SETS THE DASHED LINE FLAG
32. C C
33. NS=0
34. CALL CPLT2
35. 1000 CONTINUE
36. 1001 CONTINUE
37. CALL SVSTAT(TSA)
38. RETURN
39. END

```


HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	74	0004A
CONSTANTS:	6	00006
LOCAL VARIABLES:	5	00005
TEMPS:	1	00001

TOTAL PROGRAM: 86 00056 (PLUS LABELED COMMON)

ORIGINAL PAGE IS
OF POOR QUALITY

EXT. FORTRAN IV, VERSION F02

CPLT7

```

58 1. SUBROUTINE CPLT7
2. C THIS ROUTINE DRAWS AN AXIS AND LABELS IT.
3. COMMON /TVAL/NF1,NCTVAL(7),NM1,NCTVAL(7),INTVAL(7)
4. COMMON /ISB/ ISB,NS,IDLEH,LECH,LEHT,NFORM
5. COMMON /BND/ XMIN,XMAX,YMIN,YMAX,IBIT
6. COMMON /ABC/ A(500),B(500),C(500),Z(130)
7. COMMON /TSA/ TSA(60),INHOUS(8)
8. COMMON /VAR/XD,YD
9. COMMON /VAR2/IALPHA
10. XD=0.
11. YD=0.
12. XNAX=XMAX-XMIN
13. YNAX=YMAX-YMIN
14. IF(XMIN.LT.0) XD=-XMIN
15. IF(YMIN.LT.0) YD=-YMIN
16. XMIN=XMIN+XD
17. YMIN=YMIN+YD
18. XMAX=XMAX+XD
19. YMAX=YMAX+YD
20. CALL VWINDO(XMIN,XNAX,YMIN,YNAX)
21. IF(NFORM.EQ.2) CALL SWINDO(300,700,0,700)
22. IF(NFORM.EQ.1.AND.ISB.LT.3) CALL SWINDO(0,510,0,510)
23. IF(NFORM.EQ.1.AND.ISB.GE.3) CALL SWINDO(510,510,0,510)
24. CALL MOVEA(XMIN,YMIN)
25. CALL DRAWA(XMAX,YMAX)
26. CALL DRAWA(XMAX,YMAX)
27. CALL DRAWA(XMIN,YMAX)
28. CALL DRAWA(XMIN,YMIN)
29. GO TO 30
30. RETURN
31. END

```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC	HEX
	WORDS	WORDS
GENERATED CODE:	94	00056
CONSTANTS:	4	00004
LOCAL VARIABLES:	7	00007
TEMPS:	1	00001
TOTAL PROGRAM:	106	0006A (PLUS LABELED COMMON)

EXT. FORTRAN IV, VERSION F02

CPLT2

```

60 1. * COMPILER NOSTACK
2. SUBROUTINE CPLT2
3. COMMON /TVAL/NF1,MCTVAL(7),NN1,MCTVAL(7),INTVAL(7)
4. COMMON /ISB/ISB,NS,IOBCH,LECH,LETRA,NFORM
5. COMMON /RND/XMIN,XMAX,YMIN,YMAX,IBIT
6. COMMON /ARC/AL(500),BI(500),CI(500),ZI(100)
7. COMMON /TSA/TSA(60),INHOUS(8)
8. COMMON /VAR/XD,YD
9. COMMON /VAR2/IALPHA
10. COMMON /SCOM/KSTART,KGO,HLABEL,YLABEL,IZ,Z,IARC,N,V
11. LOGICAL HLABEL,TLABEL,TLSAVE
12. INTEGER IALPHA
13. LOGICAL FIRST
14. C RESET THE WINDOW AFTER BEING SHAPPED
15. XD=0.
16. YD=0.
17. NPTS=0
18. X RANGE=XMAX-XMIN
19. Y RANGE=YMAX-YMIN
20. IF(XMIN.LT.0) XD=-XMIN
21. IF(YMIN.LT.0) YD=-YMIN
22. XMIN=XMIN+XD
23. YMIN=YMIN+YD
24. CALL SWINDO(XMIN,IRANGE,YMIN,YRANGE)
25. IF (NFORM.EQ.1.AND.ISB.LT.3)CALL SWINDO(0,510,0,510)
26. IF (NFORM.EQ.1.AND.ISB.GE.3)CALL SWINDO(510,510,0,510)
27. IF (NFORM.EQ.2) CALL SWINDO(300,700,0,700)
28. C
29. FIRST=.TRUE.
30. KSTART=1
31. 10 CALL CSCAN
32. C KGO IS SET BY CSCAN
33. C KGO=1 NEW CONTOUR VALUE
34. C 2 NEW ARC
35. C 3 NEW SUB-ARC AFTER GAP
36. C 4 NEW POINT
37. C 5 FINISHED
38. GO TO (12,24,15,32,20),KGO
39. 12 IF(.NOT. FIRST) GO TO 24
40. FIRST=.FALSE.
41. GO TO 20
42. C GAP
43. 15 NPTS=0
44. GO TO 32
45. 20 IF(FIRST) GO TO 40
46. C TERMINATE PREVIOUS ARC
47. 24 NPTS=0
48. C LABEL END OF PREVIOUS ARC
49. IF(TLSAVE) CALL PLOT(X,Y)
50. GO TO (28,30,99,36),KGO
51. 28 IALPHA=1+MOD(IZ-1,26)
52. C LABEL BEGINNING OF NEW ARC

```

```

53. 30 TSAVE=TLABEL
54. IF(HLABEL) CALL PPLOT(X,Y)
55. C AD A POINT TO ARC
56. 32 CONTINUE
57. IALPHA=1+MOD(IZ-1,26)
58. CALL SORTUP(X,Y,NPTS)
59. XX=X
60. YY=Y
61. GO TO 10
62. C PROGRAM NEVER GOES TO 99
63. 99 CONTINUE
64. 36 CONTINUE
65. 40 RETURN
66. END

```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC	HEX
WORDS	148	00094
CONSTANTS:	5	00005
LOCAL VARIABLES:	10	0000A
TEMPS:	1	00001

TOTAL PROGRAM: 164 000A4 (PLUS LABELED COMMON)

SDNPUT

```

1. SUBROUTINE SDNPUT(XX,VV,NPTS)
2. COMMON /ISB/ISB,NS,ISLCH,LLCH,LETRA,NFORM
3. COMMON /BND/XMIN,XMAX,YMIN,YMAX
4. COMMON /STAT/STA(60),INH0US(8)
5. COMMON /VAR/XD,YD
6. COMMON /VAR2/IALPHA
7. NPTS=NPTS+1
8. XX=XX+XD
9. VV=VV+YD
10. IF(NPTS.EQ.1) GO TO 10
11. IF(NFORM.EQ.2) CALL SWINDO(300,700,0,700)
12. IF(NFORM.EQ.1.AND.ISB.LT.3) CALL SWINDO(0,510,0,510)
13. IF(NFORM.EQ.1.AND.ISB.GE.3) CALL SWINDO(510,510,0,510)
14. CALL MOVEA(XX,VV)
15. IF(NS.EQ.0) CALL DRAWA(XX,VV)
16. IF(NS.EQ.0) NN=0
17. IF(NS.EQ.0) GO TO 30
18. XX=XMAX-XMIN
19. IF(XM.GT.115.AND.NFORM.EQ.1) GO TO 20
20. IF(NS.EQ.1) CALL DASHA(XX,VV,12)
21. GO TO 30
22. CONTINUE
23. NN=NN+1
24. NTEST=MOD(NN,3)
25. IF(NTEST.EQ.0) CALL MOVEA(XX,VV)
26. IF(NTEST.EQ.1) CALL DRAWA(XX,VV)
27. IF(NTEST.EQ.2) CALL DRAWA(XX,VV)
28. CONTINUE
29. IF(NS.LT.0 .OR. NS.GT.1) OUTPUT ' NS IS NOT DEFINED CORRECTLY '
30. GO TO 10
31. VV=VV
32. RETURN
33. END

```

ORIGINAL PAGE IS
OF POOR QUALITY

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	132	00084
CONSTANTS:	7	00007
LOCAL VARIABLES:	8	00008
TEMPS:	4	00004
TOTAL PROGRAM:	151	00097 (PLUS LABELED COMMON)

ORIGINAL PAGE IS
OF POOR QUALITY

CSCAN

```

1. SUBROUTINE CSCAN
2. COMMON /ISB/ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
3. COMMON /BND/ XMIN,XMAX,YMIN,YMAX
4. COMMON /ABC/ BUFA(500),BUFB(500),BUFC(500),BUFZ(130)
5. COMMON /TSA/ TSA(60),INHOUS(8)
6. COMMON /SCOM/ KSTART,KGO,HLABEL,TLABEL,I2,I3,IARG,I7,X
7. COMMON /BLK3/ LABF
8. COMMON /IVAL/ NCTR,KTRVAL(7),NN1,NCTVAL(7),INTVAL(7)
9. C KSTART IS SET =1 BY USER TO INITIATE NEW CASE
10. C KGO=1 NEW CONTOUR VALUE
11. C 2 NEW ARC AND NEW POINT AND LABEL FLAGS
12. C 3 NEW SUB-ARC AFTER GAP, AND NEW POINT
13. C 4 NEW POINT
14. C 5 FINISHED, NO NEW POINT
15. LOGICAL HLABEL,TLABEL
16. REAL IBUFA(500),IBUFB(500),IBUFC(500)
17. INTEGER KSTART,KGO,I2,IARC
18. INTEGER JCVL,JADL,JAPL
19. REAL X,Y,Z
20. REAL BUFA,BUFB,BUFC
21. EQUIVALENCE (IBUFA,BUFA),(IBUFB,BUFB),(IBUFC,BUFC)
22. DATA LABF/32000/
23. NFI=NCTR
24. GO TO (11,15),KSTART
25. 11 KSTART=2
26. JCVL=1
27. IZ=1
28. JADL=IBUFB(JCVL)
29. IF (JADL.EQ.0) GO TO 16
30. IARC=1
31. KGO =1
32. Z=BUFA(JCVL)
33. C HERE AT BEGINING OF NEW ARC
34. 13 JAPL=ABS(IBUFA(JADL))
35. JTRY=JAPL-LABF
36. IF (JTRY.GT.0) JAPL=JTRY
37. HLABEL=(IBUFA(JADL).LT.0).AND. (IBUFA(JADL).GT.(-LABF))
38. TLABEL=(IBUFB(JADL).LT.0).AND. (IBUFB(JADL).GT.(-LABF))
39. 148 X=XMAX-BUFA(JAPL)*XMIN
40. Y=BUFB(JAPL)
41. RETURN
42. 15 JAPL=IBUFC(JAPL)
43. IF (JAPL) 151,155,152
44. C CROSSING A GAP
45. 151 JAPL=-JAPL
46. KGO=3
47. GO TO 148
48. C ORDINARY NEW POINT
49. 152 KGO=4
50. IF (X.EQ.BUFA(JAPL).AND.Y.EQ.BUFB(JAPL)) GO TO 15
51. GO TO 148
52. 155 KGO=2

```

```

66 53. JADL=IBUFG(JADL)
54. IARG=IARG+1
55. IF(JADL.GT.0) GO TO 13
56. 16 JCVL=IBUFG(JCVL)
57. 12=12+1
58. MDASH=MDL+1
59. IF(NS.EQ.1) GO TO 20
60. IF(MDASH.EQ.JCVL) NS=1
61. 20 CONTINUE
62. IF(JCVL.NE.0) GO TO 12
63. KGD=5
64. RETURN
65. END

```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC	HEX
WORDS	127	0007F
CONSTANTS	0	00000
LOCAL VARIABLES	7	00007
TEMPS	1	00001

TOTAL PROGRAM: 135 00087 (PLUS LABELED COMMON)

EXT. FORTRAN IV, VERSION F02

CONTUR

```

68 1. * COMPILER NOSTACK
2. SUBROUTINE CONTUR
3. COMMON /TVAL/ NPI,MCTVAL(7),NN1,MCTVAL(7),INTVAL(7)
4. COMMON /ISB/ISB,NS,IBLCH
5. COMMON /BND/ XMIN,XMAX,YMIN,YMAX,IBIT
6. COMMON /ABC/BUFA(500),BUFB(500),BUFC(500),Z(130)
7. COMMON /TSA/ TSA(60),INHOUS(8)
8. COMMON /CONCON/KFLAG1,KFLAG2,FUNCTM,NF,NDIM,NX,NY,NZ,NY,EXELUD
9. COMMON /AVLCON/AVAIL
10. COMMON /JUNK/IZ,LABT,KX,KY,KZ,KZ2,LABF,BIG
11. C KFLAG1 IS INITIALLY SET=1 BY USER. CONTUR RESETS KFLAG=2 OR 3,
12. C HOWEVER USER IS PERMITTED TO RESET KFLAG=3. GOOD-LOCK
13. C KFLAG=1 BEGIN NEW CASE
14. C KFLAG=2 GO TO NEXT TO CONTINUE PROCESSING
15. C KFLAG=3 CALL CTRIM TO TRIM LISTS, THEN GO TO NEXT TO
16. C CONTINUE PROCESSING
17. C KFLAG2 IS SET BY CONTUR TO BE USED IN A COMPUTED GO-TO
18. C IN THE USER PROGRAM
19. C KFLAG2=1 REQUESTING FUNCTION EVALUATION
20. C KFLAG2=2 PROCESSING INTERRUPTED. AVAILABLE SPACE USED-UP.
21. C USER SHOULD PLOT AND/OR PRINT EXISTING
22. C CONTOUR LIST AND THEN RE-ENTER CONTUR
23. C FOR FURTHER PROCESSING
24. C KFLAG2=3 PROCESSING COMPLETED. USER SHOULD PLOT
25. C AND/OR PRINT EXISTING CONTOUR LISTS
26. C KFLAG2=4 CATASTROPHIC SHORTAGE OF AVAILABLE SPACE.
27. C THE CASE CAN NOT BE COMPLETED.
28. C REAL X(5),Y(5),XCORR(2),YCORR(2),ZLO(5),ZHI(5)
29. C INTEGER SAVED,SAVE1,SAVE2,SIGN1,SIGN2,AVAIL,AVAILM
30. C INTEGER POINT1,POINT2,SWICH1,SWICH2
31. C INTEGER KZ1(4),KZ2(4),KX(4),KY(4),LAB(3)
32. C REAL IBUFA(500),IBUFB(500),IBUFC(500)
33. C INTEGER RULE(4),MODE(5),POINT,POINTR,SWITCH
34. C EQUIVALENCE (BUFA(1),IBUFA(1)),(BUFB(1),IBUFB(1)),
35. C * (BUFC(1),IBUFC(1))
36. C EQUIVALENCE (IZ,JCOL)
37. C DATA KX/1,1,2,1/
38. C DATA KY/1,1,1,2/
39. C DATA KZ1/2,2,3,0/
40. C DATA KZ2/0,3,1,1/
41. C DATA LABF/32000/,LABT/0/
42. C DATA RIG/17637/
43. C *****
44. C BRANCH ON KFLAG1
45. C GO TO (104,102,101),KFLAG1
46. C TRIM LISTS TO RETRIEVE AVAILABLE SPACE
47. C 101 CALL CTRIM(AVAIL)
48. C IF(IBUFA(AVAIL)-CT,Q) GO TO 102
49. C CATASTROPHIC SPACE SHORTAGE
50. C KFLAG2=4
51. C RETURN
52. 102 GO TO NEXT, (115,141,18,20)

```

```

53. 104 CONTINUE
54. C *****
55. IF(NF.NE.0) GO TO 106
56. C SET BRANCHES FOR NF .EQ. 0
57. ASSIGN 1165 TO LINK1
58. ASSIGN 145 TO LINK2
59. ASSIGN 831 TO LINK4
60. ASSIGN 855 TO LINK5
61. IBUFB(1)=0
62. IBUFC(1)=0
63. GO TO 111
64. C SET BRANCHES FOR NF .GT. 0
65. 106 ASSIGN 116 TO LINK1
66. ASSIGN 143 TO LINK2
67. ASSIGN 185 TO LINK4
68. ASSIGN 845 TO LINK5
69. C INITIALIZE AVAILABLE SPACE
70. NF1=NF+1
71. NDIM5=NDIM-5
72. DO 110 I=NF1,NDIM
73. IBUFA(I)=NDIM5-I
74. 110 IBUFC(I)=I+1
75. IBUFC(NDIM)=0
76. AVAIL=NF1
77. C *****
78. C INITIALIZE CVL (CONTOUR VALUE LIST)
79. DO 108 I=1,NF
80. IBUFB(I)=0
81. IBUFC(I)=I+1
82. IBUFC(NF)=0
83. C *****
84. C INITIALIZE RULE ARRAY
85. C RULE(I)=RULE TO BE USED FOR SIDE I
86. C WHERE -1=INACTIVE
87. C 0=SEARCH
88. C -1=ACTIVE
89. RULE(1)=-1
90. RULE(2)=-1
91. RULE(3)=1
92. RULE(4)=1
93. C *****
94. C SET UP X AND Y ARRAYS
95. 111 CONTINUE
96. NX1=NX-1
97. NY1=NY-1
98. STEPX=(XMAX-XMIN)/FLOAT(NX1)
99. STEPY=(YMAX-YMIN)/FLOAT(NY1)
100. XCOORD(1)=XMIN
101. YCOORD(1)=YMIN
102. XCOORD(2)=XMIN+STEPX
103. YCOORD(2)=YMIN+STEPSY
104. C SET UP INITIAL Z-ARRAY

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

105.      NZMAX=NX1+3
106.      K=1
107.      XX=XCOORD(1)
108.      YY=YCOORD(1)
109.      ASSIGN 115 TO NEXT
110.      C *****
111.      C RETURN FOR FUNCTION EVALUATION
112.      112 KFLAG2=1
113.      113 KFLAG1=2
114.      RETURN
115.      C *****
116.      115 CONTINUE
117.      GO TO LINK1,(116,1165)
118.      116 CONTINUE
119.      Z(K+2)=FUNCTN
120.      1165 CONTINUE
121.      XX=XX+STEPX
122.      K=K+1
123.      IF(K.LE.NX) GO TO 112
124.      C *****
125.      C BEGIN LOOP ON ROWS
126.      126.      118 IRON=1
127.      C RETURN HERE FROM 85+ FOR NEXT ROW OF MATRIX
128.      118 CONTINUE
129.      XX=XCOORD(1)
130.      YY=YCOORD(2)
131.      ASSIGN 141 TO NEXT
132.      GO TO 112
133.      C *****
134.      141 CONTINUE
135.      GO TO LINK2,(143,145)
136.      143 CONTINUE
137.      Z(1)=FUNCTN
138.      145 CONTINUE
139.      C *****
140.      C SET UP LOOP ON COLUMNS
141.      JCOL=1
142.      C RETURN HERE FROM 83+ FOR NEXT COLUMN OF MATRIX
143.      149 CONTINUE
144.      XX=XCOORD(2)
145.      YY=YCOORD(2)
146.      ASSIGN 18 TO NEXT
147.      GO TO 112
148.      C *****
149.      C *****
150.      18 CONTINUE
151.      GO TO LINK4,(195,831)
152.      185 CONTINUE
153.      Z(12+1)=FUNCTN
154.      C *****
155.      C SET XL0( ),ZHI( ) FOR NEW SQUARE
156.      16+6=0

```

PAGE IS
OF POOR QUALITY

```

157.      ZLO(5)=BIG
158.      ZHI(5)=BIG
159.      DO 188 ISIDE=1,4
160.      I1=K21(ISIDE)+I2
161.      I2=K22(ISIDE)+I2
162.      IF(Z(I1)-EQ-EXCLUD-OR-Z(I2)-EQ-EXCLUD) GO TO 186
163.      ZLO(ISIDE)=AMINI(Z(I1),Z(I2))
164.      ZHI(ISIDE)=AMAXI(Z(I1),Z(I2))
165.      ZLO(5)=AMINI(ZLO(5),ZLO(ISIDE))
166.      ZHI(5)=AMAXI(ZHI(5),ZHI(ISIDE))
167.      GO TO 188
168.      ZLO(ISIDE)=EXCLUD
169.      ZHI(ISIDE)=EXCLUD
170.      IEXC=IEXC+1
171.      188 CONTINUE
172.      IF(IEXC-EQ-4) GO TO 831
173.      C      SET UP LOOP ON CVL
174.      ICONT=1
175.      C      *****
176.      C      RETURN HERE FROM 831 FOR NEXT CONTUR VALUE
177.      19 CONTINUE
178.      VALUE=BUFA(IGNTR)
179.      IF((ZLO(5).GT.VALUE).OR.(VALUE.GE.ZHI(5))) GO TO 83
180.      C      *****
181.      C      TEST AVAILABLE SPACE
182.      IF(UBFA(AVAIL)-GT-0) GO TO 20
183.      C      INSUFFICIENT SPACE IN BUFFERS
184.      KFLAG1=3
185.      KFLAG2=2
186.      ASSIGN 20 TO NEXT
187.      RETURN
188.      20 CONTINUE
189.      IS=1
190.      DO 36 ISIDE=1,4
191.      IF((ZLO(ISIDE).GT.VALUE).OR.(VALUE.GE.ZHI(ISIDE))) GO TO 36
192.      IX=KX(ISIDE)
193.      IY=KY(ISIDE)
194.      I1=K21(ISIDE)+I2
195.      I2=K22(ISIDE)+I2
196.      DELZ=Z(I2)-Z(I1)
197.      FRACTN=(VALUE-Z(I1))/DELZ
198.      GO TO (32,33,33,33), ISIDE
199.      32      X(IS)=XCOR(IX)
200.      Y(IS)=YCOR(IY)+FRACTN*STEPY
201.      GO TO 35
202.      33      X(IS)=XCOR(IX)+FRACTN*STEPX
203.      Y(IS)=YCOR(IY)
204.      35      MODS(15)=RUL5(15IDE)
205.      IS=IS+1
206.      36 CONTINUE
207.      C      *****
208.      C      *****

```



```

209. C TEST FOR NUMBER OF CROSSINGS
210. GO TO (83,362,37,363,364),15
211. *****
212. C HERE FOR 1 CROSSING - SHOULD HAPPEN WITH EXCLUDED MESHPPOINTS
213. 362 MODE(2)=-1
214. X(1)=X(1)
215. Y(2)=Y(1)
216. IS=3
217. C *****
218. GO TO 37
219. C HERE FOR ERRONEOUS CROSSING COUNT
220. 363 WRITE(10,3631)
221. 3631 FORMAT(2X,25HINDICATION OF 3 CROSSINGS)
222. CALL FXEM(1000)
223. C *****
224. C HERE FOR 4 CROSSINGS
225. 364 IF(X(2).LE.X(4)) GO TO 37
226. MODE(5)=MODE(4)
227. X(5)=X(4)
228. Y(5)=Y(4)
229. MODE(4)=MODE(2)
230. X(4)=X(2)
231. Y(4)=Y(2)
232. MODE(2)=MODE(5)
233. X(2)=X(5)
234. Y(2)=Y(5)
235. C *****
236. C RETURN TO 37 FROM 71 FOR SECOND PAIR OF CROSSINGS (IF ANY)
237. 37 CONTINUE
238. LAB(1)=LABT
239. LAB(2)=LABT
240. C RETURN TO 38 FROM 545 IF SEARCH FOR P1 FAILS
241. 38 CONTINUE
242. C BRANCH ACCORDING TO MODE(1) AND MODE(2)
243. KEY=3*MODE(1)+MODE(2)+5
244. GO TO (60,42,60,43,44,43,60,42,60),KEY
245. C *****
246. C HERE IF KEY=2,8
247. C (1,S), (A,S)
248. C EXCHANGE ROLES OF POINTS 1 AND 2 AND USE 43
249. 42 MODE(5)=MODE(1)
250. X(5)=X(1)
251. Y(5)=Y(1)
252. LAB(3)=LAB(1)
253. MODE(1)=MODE(2)
254. X(1)=X(2)
255. Y(1)=Y(2)
256. LAB(1)=LAB(2)
257. MODE(2)=MODE(5)
258. X(2)=X(5)
259. Y(2)=Y(5)
260. LAB(2)=LAB(3)

```

ORIGINAL PAGE IS
OF POOR QUALITY.

```

261.      GO TO 43
262.      *****
263.      C   HERE IF KEY=4,6
264.      C   (S,I),(S,A)
265.      C   SEARCH ON P1 AND ATTACH P2
266.      43  POINT=0
267.      ASSIGN 435 TO KRTN
268.      GO TO 49
269.      CONTINUE
270.      C   ATTACH P2 TO EXISTING ARC
271.      BUFA(AVAIL)=X(2)
272.      BUFB(AVAIL)=Y(2)
273.      AVAILN=IBUFC(AVAIL)
274.      C   IF SWITCH=0--ATTACH NEW POINT TO HEAD OF ARC LIST
275.      C   IF SWITCH=1--ATTACH NEW POINT OF TAIL OF ARC LIST
276.      IF(SWITCH.NE.0) GO TO 437
277.      IBUFC(AVAIL)=POINTR
278.      BUFA(IXADL)=ISIGN(AVAIL,MODE(2))-LAB(2)
279.      GO TO 439
280.      C   HERE TO ATTACH TO TAIL LIST
281.      437  IBUFC(AVAIL)=0
282.      IBUFC(POINTR)=AVAIL
283.      IBUFB(IXADL)=ISIGN(AVAIL,MODE(2))-LAB(2)
284.      AVAIL=AVAILN
285.      GO TO 70
286.      C   *****
287.      C   HERE IF KEY=5
288.      C   (G,C)
289.      C   (S,S)
290.      C   SEARCH FOR P1 AND P2 -- JOIN THE ARCS
291.      44  POINT=0
292.      ASSIGN 445 TO KRTN
293.      GO TO 49
294.      CONTINUE
295.      POINT1=POINTR
296.      SWITCH=SWITCH
297.      IXADL1=IXADL
298.      POINT=1
299.      ASSIGN 475 TO KRTN
300.      GO TO 49
301.      CONTINUE
302.      POINT2=POINTR
303.      SWITCH2=SWITCH
304.      IXADL2=IXADL
305.      IF(IXADL1.NE.IXADL2) GO TO 4753
306.      C   COMPLETE A CLOSED CURVE
307.      C   SET FLAGS TO LABEL HEAD BUT NOT TAIL
308.      IHEAD=BUFA(IXADL1)
309.      BUFA(IXADL1)=IHEAD
310.      ITAIL=BUFB(IXADL1)
311.      IBUFC(IXADL1)=AVAIL
312.      AVAILN=IBUFC(AVAIL)

```

```

313.      IBUF8(IXADL1)=-AVAIL-LABF
314.      BUFA(AVAIL)=BUFA(IHEAD)
315.      BUFB(AVAIL)=BUFB(IHEAD)
316.      IBUFC(AVAIL)=0
317.      AVAIL=AVAILM
318.      GO TO 70
319.      4753 IF(SWCH1.EQ.SWCH2) GO TO 477
320.      IF(SWCH1.NE.0) GO TO 476
321.      C      HERE IF P1 IS HEAD, P2 IS TAIL
322.      IBUFC(POINT2)=POINT1
323.      IBUF8(IXADL2)=IBUF8(IXADL1)
324.      IREMOV=IXADL1
325.      GO TO 65
326.      C      HERE IF P1 IS TAIL, P2 IS HEAD
327.      476 IBUFC(POINT1)=POINT2
328.      IBUF8(IXADL1)=IBUF8(IXADL2)
329.      IREMOV=IXADL2
330.      GO TO 65
331.      C      HERE MUST CONNECT ARCS HEAD TO HEAD OR TAIL TO TAIL
332.      477 IF(SWCH1.EQ.0) GO TO 479
333.      C      HERE IF P1 AND P2 ARE TAILS
334.      IBUF8(IXADL2)=IBUFA(IXADL1)
335.      IBUFC(POINT2)=POINT1
336.      C      UNPACK IBUFA(IXADL1)
337.      SAVE0=ABS(IBUFA(IXADL1))
338.      SAVE1=SAVE0-LABF
339.      IF(SAVE1.GT.0) SAVE0=SAVE1
340.      SIGN1=IBUFC(SAVE0)
341.      SAVE1=ABS(SIGN1)
342.      IBUFC(SAVE0)=0
343.      GO TO 479
344.      C      HERE IF P1 AND P2 ARE HEADS
345.      478 IBUFA(IXADL2)=IBUF8(IXADL1)
346.      SAVE0=POINT1
347.      SIGN1=IBUFC(SAVE0)
348.      SAVE1=ABS(SIGN1)
349.      IBUFC(SAVE0)=POINT2
350.      C      REVERSE THE NO. 1 LIST
351.      479 SIGN2=IBUFC(SAVE1)
352.      SAVE2=ABS(SIGN2)
353.      IBUFC(SAVE1)=ISIGN(SAVE0,SIGN1)
354.      IF(SAVE2.EQ.0) GO TO 479I
355.      SAVE0=SAVE1
356.      SAVE1=SAVE2
357.      SIGN1=SIGN2
358.      GO TO 479
359.      479I IREMOV=IXADL1
360.      GO TO 65
361.      C      *****
362.      C      SEARCH FOR POINT ON APL (ARC-POINT-LIST)
363.      49 IXADL=IBUF8(ICNTR)
364.      KNEW=X(POINT+1)

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

365.      YNEW=Y(POINT+1)
366.      50 IF(IXADL-EQ-0) GO TO 54
367.      IHEAD=IBUFA(IXADL)
368.      IFAIL=IBUFB(IXADL)
369.      POINTR=IHEAD
370.      SWITCH=0
371.      GO TO 52
372.      51 POINTR=ITAIL
373.      SWITCH=1
374.      C EXAMINE ONLY ACTIVE ENDS OF ARCS
375.      52 IF(POINTR-LE-0) GO TO 53
376.      XOLD=BUFA(POINTR)
377.      YOLD=IBUFB(POINTR)
378.      C COMPARE POINTS (XNEW,YNEW) AND (XOLD,YOLD)
379.      IF((XNEW-NE-XOLD).OR.(YNEW-NE-YOLD)) GO TO 53
380.      IF(POINT-EQ-0) GO TO 525
381.      IF(POINT1-EQ-POINTR) GO TO 53
382.      KFOUND=1
383.      GO TO 55
384.      53 IF(SWITCH-EQ-0) GO TO 51
385.      IXADL=IBUFC(IXADL)
386.      GO TO 50
387.      54 KFOUND=0
388.      C *****
389.      C SEARCH FAILED
390.      IF(POINT-NE-0) GO TO 545
391.      MODE(1)=-1
392.      LAB(1)=LABF
393.      GO TO 38
394.      545 MODE(2)=-1
395.      LAB(2)=LABF
396.      KEY=4
397.      SWITCH=SWTCH1
398.      POINTR=POINT1
399.      IXADL=IXADL1
400.      GO TO 435
401.      55 CONTINUE
402.      GO TO KRTN,(435,445,475)
403.      C *****
404.      C HERE IF KEY=1,3,7,9
405.      C (1,1),(1,A),(A,1),(A,A)
406.      C HERE TO START NEW ARC
407.      C FIRST OBTAIN 3 TRIPLE WORDS FROM AVAILABLE LIST
408.      60 IXADL=AVAIL
409.      IHEAD=IBUFC(IXADL)
410.      IFAIL=IBUFC(IHEAD)
411.      AVAIL=IBUFC(ITAIL)
412.      C THEN CONSTRUCT NEW ARC
413.      BUFA(IHEAD)=X(1)
414.      BUFB(IHEAD)=Y(1)
415.      IBUFC(IHEAD)=ITAIL
416.      BUFA(ITAIL)=X(2)

```

76

```
417.      BUFB(ITAIL)=Y(2)
418.      IBUFC(ITAIL)=0
419.      IBUFA(IXADL)=ISIGN(IHEAD,MODE(1))-LAB(1)
420.      IBUFB(IXADL)=ISIGN(IFAIF,MODE(2))-LAB(2)
421.      IBUFC(IXADL)=IBUFB(ICNTR)
422.      IBUFB(ICNTR)=IXADL
423.      GO TO 70
424.      C *****
425.      C HERE TO RESTORE A WORD TO AVAILABLE SPACE
426.      C CONTINUE
427.      C FIRST, EXAMINE CVL LIST
428.      IXADL=IBUFB(ICNTR)
429.      IF(IXADL.NE.IREMOV) GO TO 653
430.      IBUFB(ICNTR)=IBUFC(IREMOV)
431.      GO TO 655
432.      IXADL=IBUFC(IXADL)
433.      IF(IREMOV(IXADL).NE.IREMOV) GO TO 652
434.      IBUFC(IXADL)=IREMOV(IREMOV)
435.      IBUFC(IREMOV)=AVAIL
436.      IBUFA(IREMOV)=IBUFA(AVAIL)+1
437.      AVAIL=IREMOV
438.      C *****
439.      C CONTINUE
440.      IF(IS.LT.5) GO TO 83
441.      DO 71 I=1,2
442.      MODE(I)=MODE(I+2)
443.      X(I)=X(I+2)
444.      Y(I)=Y(I+2)
445.      IS=3
446.      GO TO 37
447.      C *****
448.      C HERE AFTER PROCESSING ONE SQUARE FOR A SINGLE CNTR *****
449.      C CONTINUE
450.      ICNTR=ICNTR+1
451.      IF(ICNTR.LE.NF) GO TO 19
452.      C CONTINUE
453.      XCOORD(1)=XCOORD(2)
454.      XCOORD(2)=XCOORD(2)+STEPX
455.      JCUL=JCUL+1
456.      RULR(1)=0
457.      IF(JCUL-NX1) 149,833,84
458.      RULR(2)=-1
459.      GO TO 149
460.      C *****
461.      C HERE AFTER ALL COLUMNS OF MATRIX HAVE BEEN PROCESSED *****
462.      C MOVE Z-VALUES DOWN 2 POSITIONS AND INCREMENT YCOORD(1)
463.      C YCOORD(2), FOR NEXT ROW
464.      C CONTINUE
465.      GO TO LINK5,(845,855)
466.      C CONTINUE
467.      DO 85 K=3,NZMAX
468.      KBACK=3+NZMAX-K
```

```

469. 85 Z(KBACK)=Z(KBACK-2)
470. 855 CONTINUE
471. YCOOR(1)=YCOOR(2)
472. YCOOR(2)=YCOOR(2)+STEPY
473. XCOOR(1)=XMIN
474. XCOOR(2)=XMIN+STEPX
475. IROW=IROW+1
476. RULE(1)=-1
477. RULE(2)=0
478. RULE(3)=+1
479. IF(IROW-NY1) 118,853,86
480. 863 RULE(4)=-1
481. GO TO 118
482. C *****
483. C *****
484. C *****
485. 86 HERE WHEN FINISHED WITH ALL CONTOURS FOR ENTIRE MAT.
486. KFLAG2=3
487. KFLAG1=1
488. RETURN
489. END

```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC	HEX
WORDS	1019	003FB

GENERATED CODE:	935	00390
CONSTANTS:	2	00002
LOCAL VARIABLES:	90	0005A
TEMPS:	2	00002

TOTAL PROGRAM: 1019 003FB (PLUS LABELED COMMON)

EXT. FORTRAN IV, VERSION F02

PTIME

```
80 1. SUBROUTINE PTIME
2. DIMENSION MIA(64),NBT(4),NF(10),ID(4)
3. COMMON /HEAD/ INPBUF(512)
4. COMMON /IDAT/IDT(4),NZFT,MENN,NSXP,NPOL,NPOS,YPOS
5. INTEGER DT
6. DATA NBT/7,7,6,10/
7. DATA NF /1,2,4,8,10,20,40,80,100,200/
8. DATA ID /227F,227F,223F,323F/
9. DATA KK/42FFFF/
10. DATA MNW /1/
11. DATA KZF/421FFF/
12. DATA KSH/22FF/
13. DATA KMIA/82FFFF0000/
14. DATA KEXP/127/
15. IWD = 1
16. DO 50 J1=1,64
17. MIA(J1) = INPBUF(IWD)
18. IWD = IWD + 1
19. 50 CONTINUE
20. IDAT = ISL(MIA(1),-16)+IAND(ISL(MIA(1),16),KMIA)
21. DO 252 I1=1,4
22. IF (I1.EQ.1) NBB = 0
23. IF (I1.GT.1) NBB = NBT(I1-1) + NBB
24. DT(I1) = IAND(ID(I1),ISL(IDAT,-NBB))
25. 252 CONTINUE
26. NZFT = IAND(ISL(MIA(2),-16),KZF)
27. MENW = IAND(ISL(MIA(2),-8),KSH)
28. NEXP = IAND(ISL(MIA(3),-25),KEXP)
29. NPOL = IAND(ISL(MIA(3),-19),7)
30. DO 25 I1=1,8
31. 25 CONTINUE
32. IF(NEXP.EQ.2) NEXP=60
33. IF(NEXP.EQ.3) NEXP=30
34. IF(NEXP.EQ.4) NEXP=15
35. IF(NEXP.EQ.5) NEXP=8
36. IIX = ISL(MIA(3),-16)+IAND(ISL(MIA(3),16),KMIA)
37. ISNGX = IAND(ISL(IIX,-13),1)
38. XPOS = 0
39. TXPOS = 0
40. MX = 14
41. C DO 333 I1=1,3
42. I1=3
43. 333 CONTINUE
44. MN = 4
45. 353 CONTINUE
46. TXPOS = TXPOS + IAND(ISL(IIX,MX),1)*(2**(MN-1))
47. MN = MN - 1
48. MX = MX + 1
49. IF(MN.GE.1) GO TO 353
50. XPOS = XPOS + TXPOS*(10**(I1-1))
51. TXPOS = 0
52. IF(I1.LE.1) GO TO 363
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
53.      II=II-1
54.      GO TO 333
55.      363 CONTINUE
56.      IF (ISNGR.EQ.0) XPOS=XPOS
57.      ISNGV=IAND(ISL(IXX,-MX),1)
58.      XPOS=0
59.      TVPOS=0
60.      MX=MX+1
61.      NN=4
62.      454 CONTINUE
63.      TVPOS=TVPOS+(IAND(ISL(IXX,-MX),1))*2**((NN-1)
64.      NN=NN-1
65.      MX=MX+1
66.      IF (NN.GE.1) GO TO 454
67.      YPOS=YPOS+TVPOS
68.      YPOS=YPOS*(10**2)
69.      TVPOS=0
70.      MV=23
71.      C      DO 444 II=1,2
72.      II=2
73.      444 CONTINUE
74.      NN=4
75.      464 CONTINUE
76.      IYV=ISL(MIA(4),-16)*IAND(ISL(MIA(4),16),MIA)
77.      TVPOS=TVPOS + (IAND(ISL(IYV,-MV),1))*2**((NN-1)
78.      NN=NN-1
79.      MV = MV + 1
80.      IF (NN.GE.1) GO TO 464
81.      YPOS=YPOS + TVPOS*(10**((II-1))
82.      TVPOS=0
83.      C 444 CONTINUE
84.      IF (II.LE.1) GO TO 474
85.      II=II-1
86.      GO TO 444
87.      474 CONTINUE
88.      IF (ISNGY.EQ.0) YPOS=YPOS
89.      XPOS=XPOS/100.
90.      YPOS=YPOS/100.
91.      RETURN
92.      END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
--	--------------	--------------

GENERATED CODE:	267	00108
-----------------	-----	-------

CONSTANTS:	5	00005
------------	---	-------

LOCAL VARIABLES:	104	00068
------------------	-----	-------

TEMPS:	4	00004
--------	---	-------

TOTAL PROGRAM:	380	0017C (PLUS LABELED COMMON)
----------------	-----	-----------------------------

APPENDIX C
COMMAND FILES FOR
KZTIME, KZLONGT, AND KZTRANSV

C CLKZTIME
!ECHO
!SET F113 FTOTEST
!SET F110 ME
!C CLKZTIME, CLKZTIME, BIT OVER BITEMP
!LYNX BITEMP OVER CLKZTIME(SL,F) (N)

!C CLKZTIME
!ECHO
!ECHO
!SET F113 FTOTEST
!SET F110 ME
!CLKZTIME.

APPENDIX D
DESCRIPTIONS FOR PROGRAM SUBROUTINES

KZTIME: To read the 9TRK data tape and search for the beginning record of each file and also print out the housekeeping information for each file.

KZLONGT: The main program for longitudinal plot.
To read data file 1A1B from 9TRK data tape, compute contour data $(1A - 1B)/(1A + 1B)$ and save it for future use.
To read information from terminal input and call subroutines to do contour plots.

KZTRANSV: The main program for transverse.
To read input data 2A2B, 3A3B, compute contour data, azimuth data and plot.

KZPTIME: To generate housekeeping information.

KZFILTER: To generate filter data.

KZDRIV 1: To choose saved data file for different cases, plot longitudinal contour and intensity plot and also display the label with the plots.

KZSMPLT: To set up X, Y range, clear screen and call subroutines to plot save status words.

KZCONTUR: Generate contour plot points.

KZCPLT 1: Draw plotting frame.

KZCPLT 2: To decide starting or ending contour and call subroutine KZSDNPUT to plot.

KZSDNPUT: To plot contour.

KZPPLOT: Options 1. Delete all characters not in box.
2. Delete all characters.
3. Delete only low level characters.

KZCSCAN: Set up KGO
KGO = 1 New contour value
KGO = 2 New ARC
KGO = 3 New SUB-ARC After gap

KGO = 4 New point

KGO = 5 Finished

APPROVAL

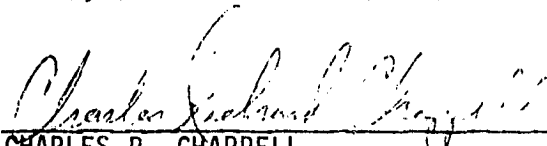
LONGITUDINAL AND TRANSVERSE MAGNETIC FIELD PROGRAM PROCEDURE
AND DETAILED SPECIFICATION (FOR SIGMA 5)

By Caroline K. Wang

The information in this report has been reviewed for technical content. Review of any information concerning Department of Defense or nuclear energy activities or programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.



ANTHONY C. DeLOACH
Chief, Solar Sciences Branch



CHARLES R. CHAPPELL
Chief, Solar-Terrestrial Division



CHARLES A. LUNDQUIST
Director, Space Sciences Laboratory

ORIGINAL FILED IS
OF POOR QUALITY